

Metody Sztucznej Inteligencji

Wykład 3: **Już nie leniwe, ale naiwne**

dr inż. Paweł Ksieniewicz
Katedra Systemów i Sieci Komputerowych

3 kwietnia 2019



Regresja liniowa



Czym jest regresja liniowa?

- ↪ Próbuje modelować zależność pomiędzy dwiema zmiennymi losowymi, dopasowując równanie liniowe do zbioru obserwacji.

Czym jest regresja liniowa?

↪ Próbuje modelować zależność pomiędzy dwiema zmiennymi losowymi, dopasowując równanie liniowe do zbioru obserwacji.

Prosta regresja liniowa.

Czym jest regresja liniowa?

- ↪ Próbuje modelować zależność pomiędzy dwiema zmiennymi losowymi, dopasowując równanie liniowe do zbioru obserwacji.

Prosta regresja liniowa.

- ↪ Wartość **zmiennej objaśnianej** (zależnej) staramy się wyznaczyć (określając jej zależność) na podstawie **zmiennej objaśniającej** (niezależnej).

Czym jest regresja liniowa?

- ↪ Próbuje modelować zależność pomiędzy dwiema zmiennymi losowymi, dopasowując równanie liniowe do zbioru obserwacji.

Prosta regresja liniowa.

- ↪ Wartość **zmiennej objaśnianej** (zależnej) staramy się wyznaczyć (określając jej zależność) na podstawie **zmiennej objaśniającej** (niezależnej).

Nie stosujemy tego pojęcia, bo jest mylące z zależnością statystyczną.

Czym jest regresja liniowa?

- ↪ Próbuje modelować zależność pomiędzy dwiema zmiennymi losowymi, dopasowując równanie liniowe do zbioru obserwacji.

Prosta regresja liniowa.

- ↪ Wartość **zmiennej objaśnianej** (zależnej) staramy się wyznaczyć (określając jej zależność) na podstawie **zmiennej objaśniającej** (niezależnej).

Nie stosujemy tego pojęcia, bo jest mylące z zależnością statystyczną.

- ↪ Pozwala nam na:

- × Określenie relacji pomiędzy dwiema zmiennymi losowymi.

Czym jest regresja liniowa?

↪ Próbuje modelować zależność pomiędzy dwiema zmiennymi losowymi, dopasowując równanie liniowe do zbioru obserwacji.

Prosta regresja liniowa.

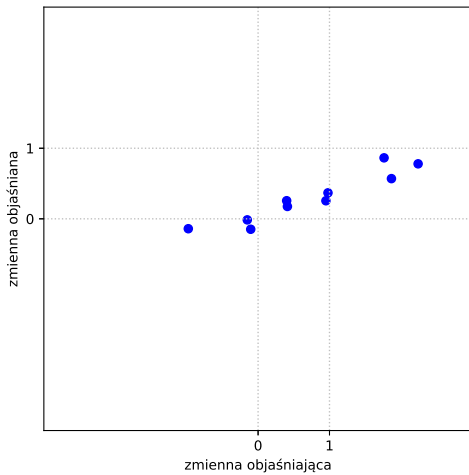
↪ Wartość **zmiennej objaśnianej** (zależnej) staramy się wyznaczyć (określając jej zależność) na podstawie **zmiennej objaśniającej** (niezależnej).

Nie stosujemy tego pojęcia, bo jest mylące z zależnością statystyczną.

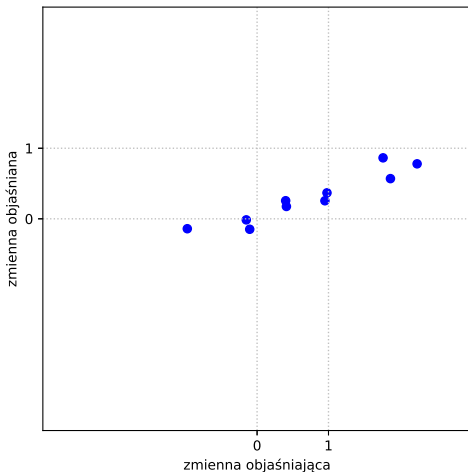
↪ Pozwala nam na:

- × Określenie relacji pomiędzy dwiema zmiennymi losowymi.
- × Określenie czy dwie zmienne losowe są od siebie istotnie zależne statystycznie.

Regresja liniowa

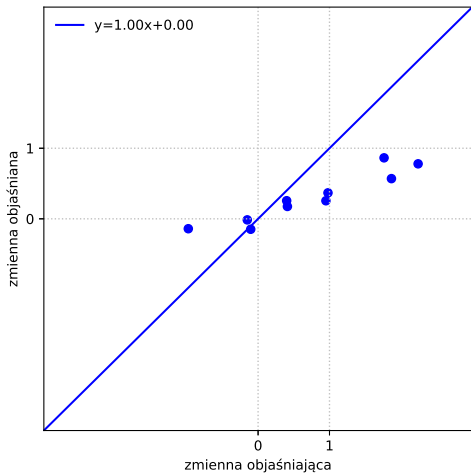


Regresja liniowa

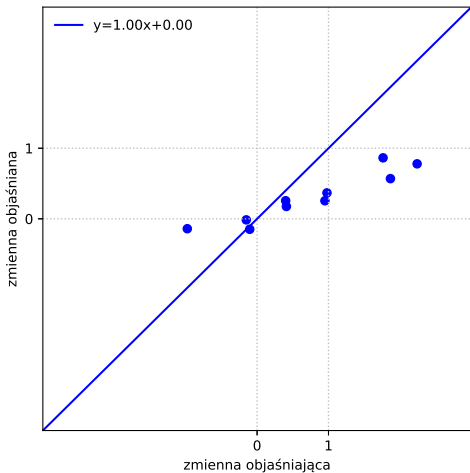


→ Załóżmy, że zmienną objaśniającą będzie znormalizowana liczba godzin spędzonych w pracy, a objaśnianą znormalizowana liczba wypalanych dziennie papierosów.

Regresja liniowa — dopasowanie prostej



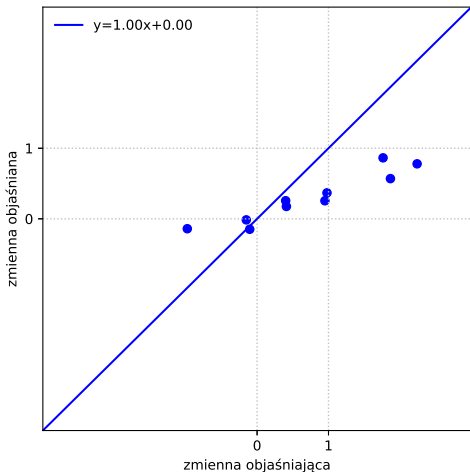
Regresja liniowa — dopasowanie prostej



↪ Linia najlepszego dopasowania.

$$y = ax + b$$

Regresja liniowa — dopasowanie prostej

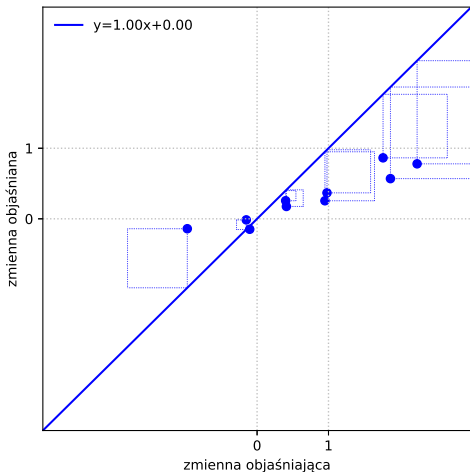


↪ Linia najlepszego dopasowania.

$$y = ax + b$$

- × a — współczynnik kierunkowy (slope)
- × b — wyraz wolny (intercept)

Regresja liniowa — dopasowanie prostej



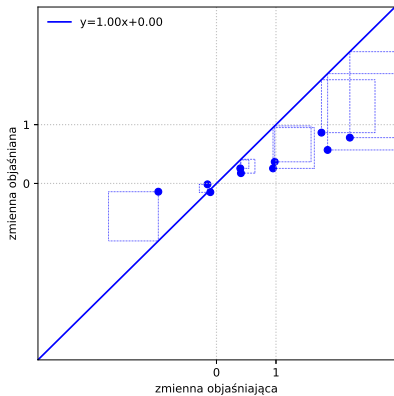
↪ Linia najlepszego dopasowania.

$$y = ax + b$$

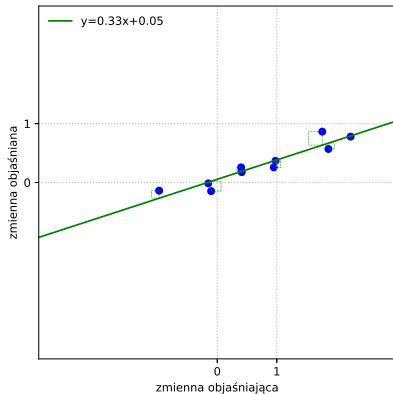
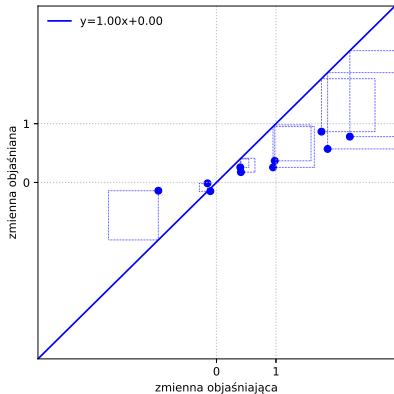
- × a — współczynnik kierunkowy (slope)
- × b — wyraz wolny (intercept)

```
def linear_regression(x, y):  
    n = len(y)  
    mean_x, mean_y = np.mean(x), np.mean(y)  
  
    SS_xy = np.sum(x * y) - n * mean_y * mean_x  
    SS_xx = np.sum(x * x) - n * mean_x * mean_x  
  
    a = SS_xy / SS_xx  
    b = mean_y - a * mean_x  
  
    return a, b
```

Regresja liniowa — metoda najmniejszych kwadratów



Regresja liniowa — metoda najmniejszych kwadratów



Charakter zmiennej objaśniającej

Charakter zmiennej objaśniającej

↷ $a > 0$: dodatnia korelacja ze zmienną objaśnianą
(**stymulanta**),

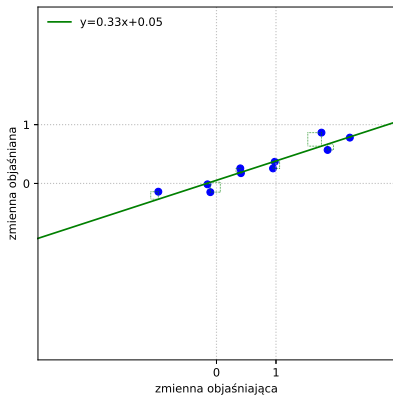
Charakter zmiennej objaśniającej

- ↷ $a > 0$: dodatnia korelacja ze zmienną objaśnianą (**stymulanta**),
- ↷ $a < 0$: ujemna korelacja ze zmienną objaśnianą (**destymulanta**),

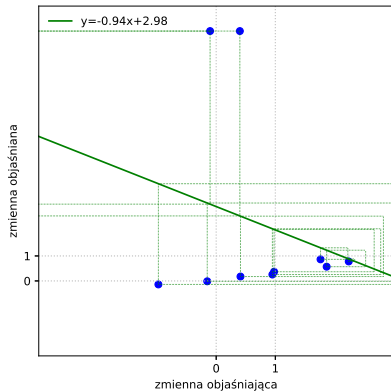
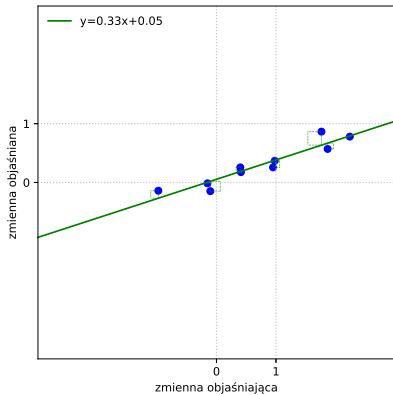
Charakter zmiennej objaśniającej

- ↪ $a > 0$: dodatnia korelacja ze zmienną objaśnianą (**stymulanta**),
- ↪ $a < 0$: ujemna korelacja ze zmienną objaśnianą (**destymulanta**),
- ↪ $a = 0$: zmienna niezależna od zmiennej objaśnianej.

Regresja liniowa



Regresja liniowa





Regresja logistyczna



Czym jest regresja logistyczna?

- Model regresji logistycznej przyjmuje rzeczywiste wejścia i dokonuje predykcji prawdopodobieństwa przynależności do klasy.

Czym jest regresja logistyczna?

- ↪ Model regresji logistycznej przyjmuje rzeczywiste wejścia i dokonuje predykcji prawdopodobieństwa przynależności do klasy.
- ↪ Jest to klasyfikator binarny.

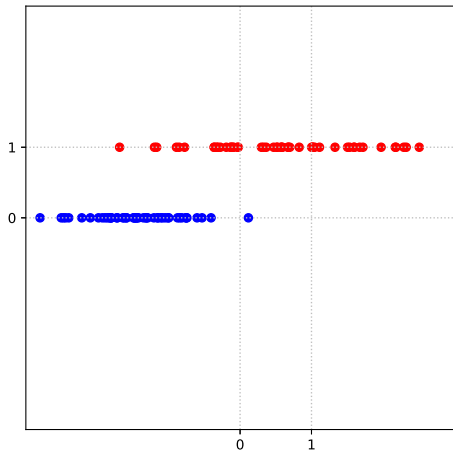
Czym jest regresja logistyczna?

- ☞ Model regresji logistycznej przyjmuje rzeczywiste wejścia i dokonuje predykcji prawdopodobieństwa przynależności do klasy.
- ☞ Jest to klasyfikator binarny.
- ☞ To, że nazywa się regresją, nie znaczy, że służy do rozwiązywania zadania regresji.

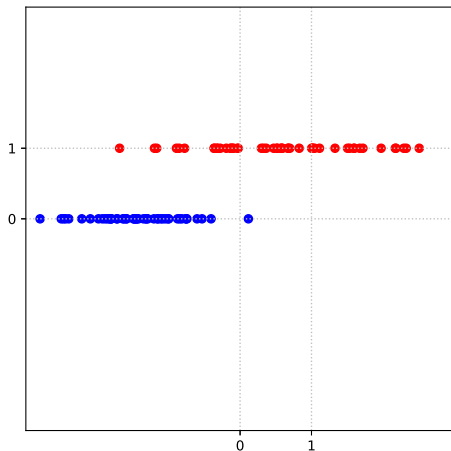
Czym jest regresja logistyczna?

- ↪ Model regresji logistycznej przyjmuje rzeczywiste wejścia i dokonuje predykcji prawdopodobieństwa przynależności do klasy.
- ↪ Jest to klasyfikator binarny.
- ↪ To, że nazywa się regresją, nie znaczy, że służy do rozwiązywania zadania regresji.
- ↪ Nie mamy dla niej, jak dla regresji liniowej, prostego równania, a nareszcie musimy **dopasować model**.

Regresja logistyczna

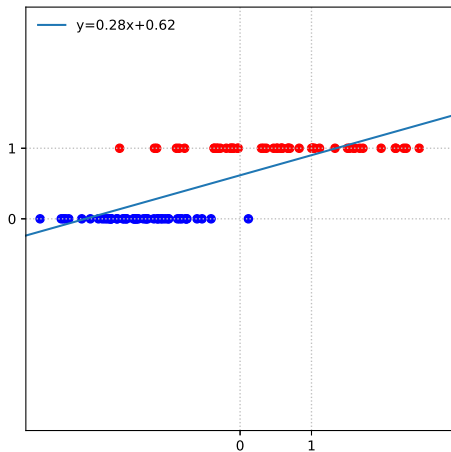


Regresja logistyczna

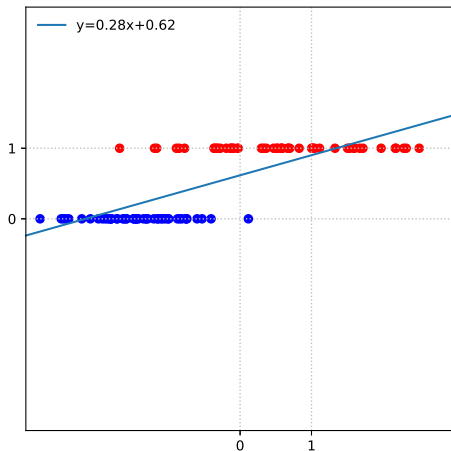


⇒ Tym razem
dysponujemy zbiorem
obserwacji z dy-
chotomiczną funkcją celu.

Regresja logistyczna

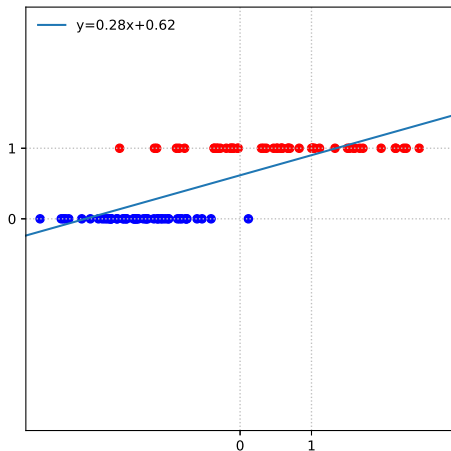


Regresja logistyczna



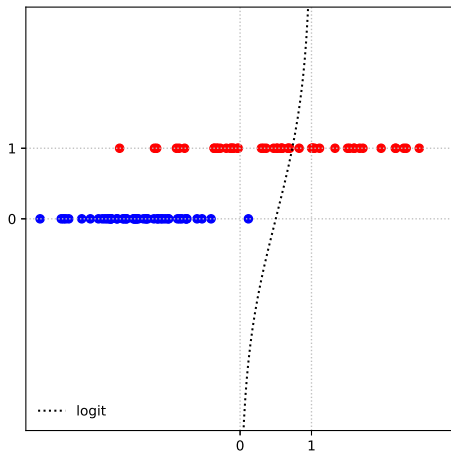
⇒ Teoretycznie
moglibyśmy skorzystać
tutaj z regresji
liniowej.

Regresja logistyczna

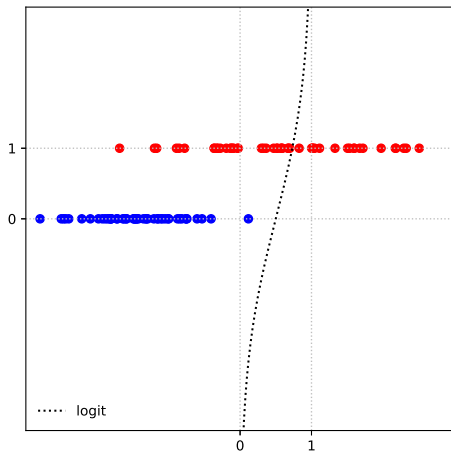


- ↪ Teoretycznie moglibyśmy skorzystać tutaj z regresji liniowej.
- ↪ Dlaczego nie?

Regresja logistyczna

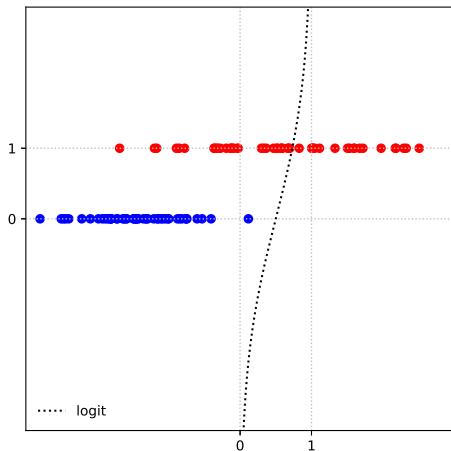


Regresja logistyczna



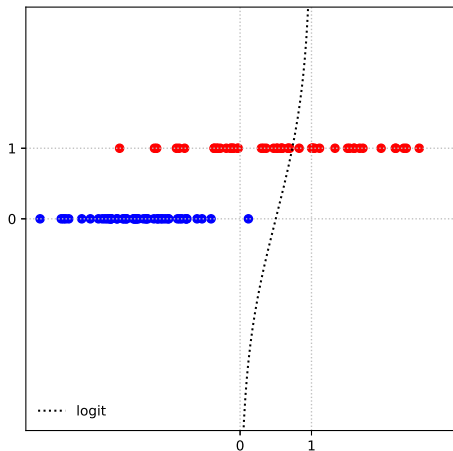
♀→ W 1934 Bliss wymyślił funkcję *probit*. Nie przyjęła się

Regresja logistyczna



♀→ W 1934 Bliss wymyślił funkcję *probit*. Nie przyjęła się ze względu na złożoność obliczeniową.

Regresja logistyczna

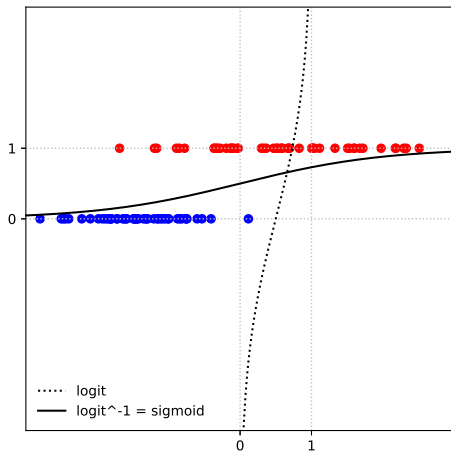


♀→ W 1934 Bliss wymyślił funkcję *probit*. Nie przyjęła się ze względu na złożoność obliczeniową.

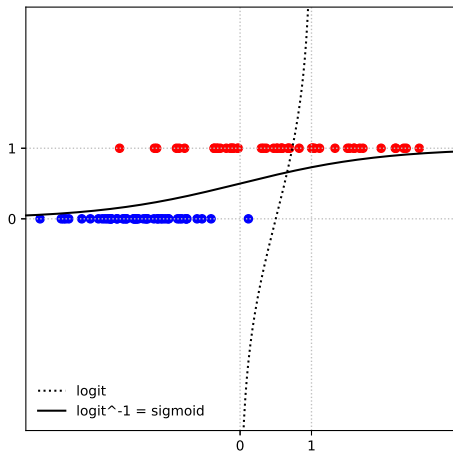
♀→ W 1944 Berkson wpadł na *logit*.

$$\text{logit}(z) = \log \frac{z}{1-z}$$

Regresja logistyczna



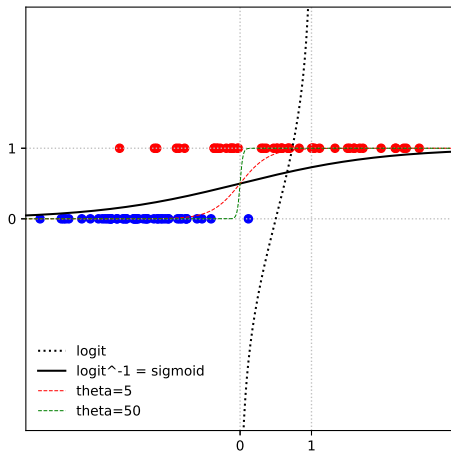
Regresja logistyczna



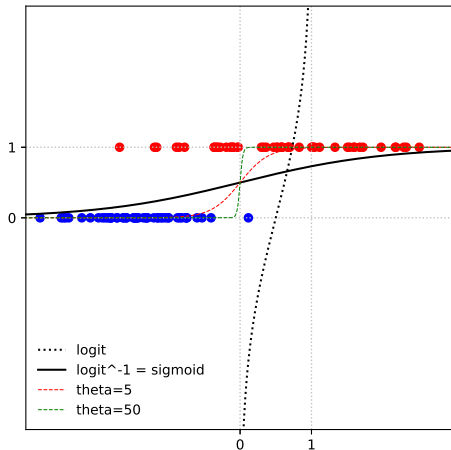
↪ Funkcja sigmoidalna

$$\text{logit}^{-1}(z) = \frac{1}{1 + e^{-z}}$$

Regresja logistyczna

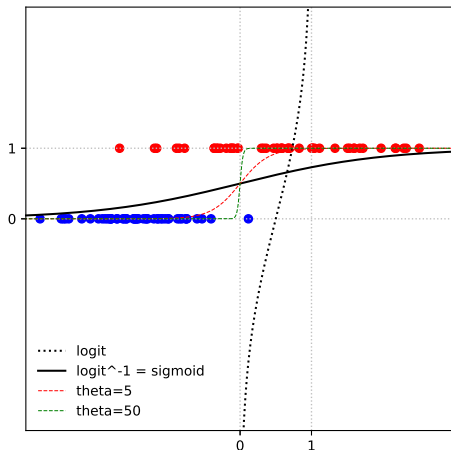


Regresja logistyczna



⇒ Tym razem dopasowujemy krzywą wyznaczaną przez funkcję sigmoidalną.

Regresja logistyczna



- ⇒ Tym razem dopasowujemy krzywą wyznaczaną przez funkcję sigmoidalną.
- ⇒ Poszukujemy takiego parametru θ , będącego mnożnikiem zmiennej objaśniającej, który zminimalizuje różnicę pomiędzy predykcją i zmienną objaśnianą.

Metoda gradientu prostego w regresji logistycznej

- 1 Wybierz startową wartość θ :

Metoda gradientu prostego w regresji logistycznej

- 1 Wybierz startową wartość θ :

$$\theta = 0$$

Metoda gradientu prostego w regresji logistycznej

- 1 Wybierz startową wartość θ :

$$\theta = 0$$

- 2 Wyznacz h – wartość funkcji sigmoidalnej dla iloczynu skalarnego zbioru uczącego i θ :

Metoda gradientu prostego w regresji logistycznej

- 1 Wybierz startową wartość θ :

$$\theta = 0$$

- 2 Wyznacz h – wartość funkcji sigmoidalnej dla iloczynu skalarnego zbioru uczącego i θ :

$$h = X \cdot \theta$$

Metoda gradientu prostego w regresji logistycznej

- 1 Wybierz startową wartość θ :

$$\theta = 0$$

- 2 Wyznacz h – wartość funkcji sigmoidalnej dla iloczynu skalarnego zbioru uczącego i θ :

$$h = X \cdot \theta$$

- 3 Wylicz *gradient* jako średnią wartość iloczynu skalarnego zbioru uczącego i różnicy pomiędzy h a zbiorem etykiet:

Metoda gradientu prostego w regresji logistycznej

- 1 Wybierz startową wartość θ :

$$\theta = 0$$

- 2 Wyznacz h – wartość funkcji sigmoidalnej dla iloczynu skalarnego zbioru uczącego i θ :

$$h = X \cdot \theta$$

- 3 Wylicz *gradient* jako średnią wartość iloczynu skalarnego zbioru uczącego i różnicy pomiędzy h a zbiorem etykiet:

$$\text{gradient} = \frac{X \cdot (h - y)}{n}$$

Metoda gradientu prostego w regresji logistycznej

- 1 Wybierz startową wartość θ :

$$\theta = 0$$

- 2 Wyznacz h – wartość funkcji sigmoidalnej dla iloczynu skalarnego zbioru uczącego i θ :

$$h = X \cdot \theta$$

- 3 Wylicz *gradient* jako średnią wartość iloczynu skalarnego zbioru uczącego i różnicy pomiędzy h a zbiorem etykiet:

$$\text{gradient} = \frac{X \cdot (h - y)}{n}$$

- 4 Zredukuj θ o iloczyn *gradientu* i współczynnika kroku:

Metoda gradientu prostego w regresji logistycznej

- 1 Wybierz startową wartość θ :

$$\theta = 0$$

- 2 Wyznacz h – wartość funkcji sigmoidalnej dla iloczynu skalarnego zbioru uczącego i θ :

$$h = X \cdot \theta$$

- 3 Wylicz *gradient* jako średnią wartość iloczynu skalarnego zbioru uczącego i różnicy pomiędzy h a zbiorem etykiet:

$$\text{gradient} = \frac{X \cdot (h - y)}{n}$$

- 4 Zredukuj θ o iloczyn *gradientu* i współczynnika kroku:

$$\theta = \theta - \text{gradient} \cdot lr$$

Metoda gradientu prostego w regresji logistycznej

- 1 Wybierz startową wartość θ :

$$\theta = 0$$

- 2 Wyznacz h – wartość funkcji sigmoidalnej dla iloczynu skalarnego zbioru uczącego i θ :

$$h = X \cdot \theta$$

- 3 Wylicz *gradient* jako średnią wartość iloczynu skalarnego zbioru uczącego i różnicy pomiędzy h a zbiorem etykiet:

$$gradient = \frac{X \cdot (h - y)}{n}$$

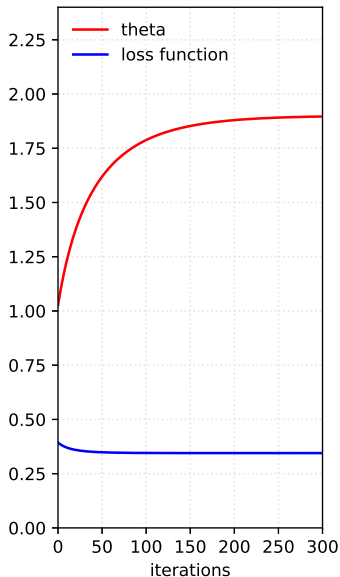
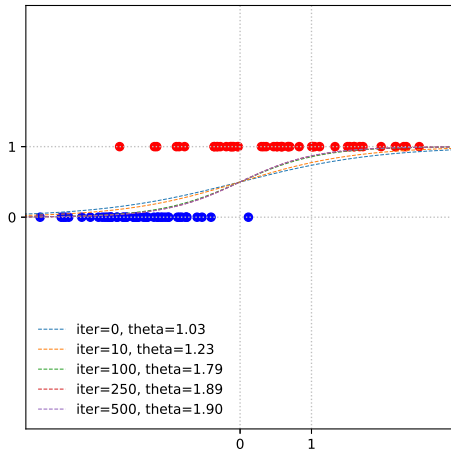
- 4 Zredukuj θ o iloczyn *gradientu* i współczynnika kroku:

$$\theta = \theta - gradient \cdot lr$$

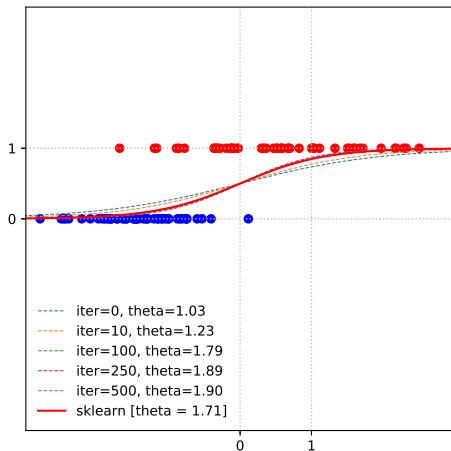
- 5 Wróć do 2 jeśli nie wyczerpał się limit iteracji.

```
def sigmoid(z):  
    return 1 / (1 + np.exp(-z))  
  
def logistic_regression(X, y):  
    num_iter = 500  
    learning_rate = 0.2  
  
    theta = np.ones(X.shape[1])  
    for i in range(num_iter):  
        z = np.dot(X, theta)  
        h = sigmoid(z)  
  
        gradient = np.dot(X.T, (h - y)) / y.size  
        theta -= learning_rate * gradient  
  
    return theta
```

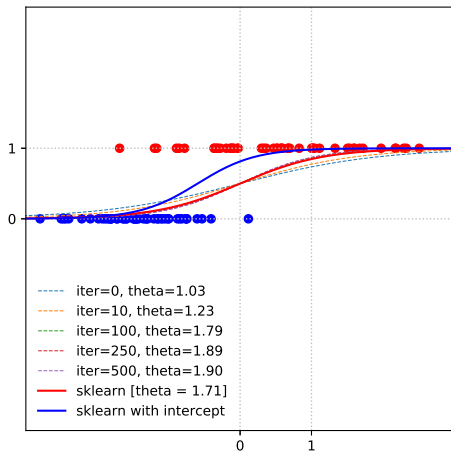
Regresja logistyczna



Regresja logistyczna



Regresja logistyczna



Regresja logistyczna

↪ Zmienna objaśniana musi być dychotomiczna (klasyfikacja binarna).

Regresja logistyczna

- ↪ Zmienna objaśniana musi być dychotomiczna (klasyfikacja binarna).
- ↪ Zakłada brak błędów pomiaru, a więc obserwacje odstające silnie wpływają na jej wynik.

Regresja logistyczna

- ↪ Zmienna objaśniana musi być dychotomiczna (klasyfikacja binarna).
- ↪ Zakłada brak błędów pomiaru, a więc obserwacje odstające silnie wpływają na jej wynik.
- ↪ Warto pozbyć się silnie skorelowanych ze sobą cech, aby uniknąć przeuczenia.

Regresja logistyczna

- ↪ Zmienna objaśniana musi być dychotomiczna (klasyfikacja binarna).
- ↪ Zakłada brak błędów pomiaru, a więc obserwacje odstające silnie wpływają na jej wynik.
- ↪ Warto pozbyć się silnie skorelowanych ze sobą cech, aby uniknąć przeuczenia.
- ↪ Potrafi nie dojść do żadnych rozsądnych wniosków (nie osiągnąć konwergencji).



Naiwny klasyfikator bayesowski



Twierdzenie Bayesa

$$P(h|x) = \frac{P(x|h) * P(h)}{P(x)}$$

Twierdzenie Bayesa

$$P(h|x) = \frac{P(x|h) * P(h)}{P(x)}$$

↗ $P(h)$ — prawdopodobieństwo zajścia hipotezy h (a priori),

Twierdzenie Bayesa

$$P(h|x) = \frac{P(x|h) * P(h)}{P(x)}$$

- ↗ $P(h)$ — prawdopodobieństwo zajścia hipotezy h (a priori),
- ↗ $P(x)$ — prawdopodobieństwo x ,

Twierdzenie Bayesa

$$P(h|x) = \frac{P(x|h) * P(h)}{P(x)}$$

- ↗ $P(h)$ — prawdopodobieństwo zajścia hipotezy h (a priori),
- ↗ $P(x)$ — prawdopodobieństwo x ,
- ↗ $P(x|h)$ — prawdopodobieństwo wystąpienia x , kiedy zajdzie h ,

Twierdzenie Bayesa

$$P(h|x) = \frac{P(x|h) * P(h)}{P(x)}$$

- ↗ $P(h)$ — prawdopodobieństwo zajścia hipotezy h (a priori),
- ↗ $P(x)$ — prawdopodobieństwo x ,
- ↗ $P(x|h)$ — prawdopodobieństwo wystąpienia x , kiedy zajdzie h ,
- ↗ $P(h|x)$ — prawdopodobieństwo a posteriori.

Naiwny klasyfikator bayesowski (i)

- ↪ Celujemy w wyliczenie prawdopodobieństwa a posteriori ($P(h|x)$) na podstawie prawdopodobieństwa a priori ($P(h)$) oraz $P(x)$ i $P(x|h)$.

Naiwny klasyfikator bayesowski (i)

- ↪ Celujemy w wyliczenie prawdopodobieństwa a posteriori ($P(h|x)$) na podstawie prawdopodobieństwa a priori ($P(h)$) oraz $P(x)$ i $P(x|h)$.
- ↪ Algorytm Naive Bayes to prosta metoda wykorzystująca prawdopodobieństwo przynależności każdej z cech wzorca do klasy.

Naiwny klasyfikator bayesowski (i)

- ↪ Celujemy w wyliczenie prawdopodobieństwa a posteriori ($P(h|x)$) na podstawie prawdopodobieństwa a priori ($P(h)$) oraz $P(x)$ i $P(x|h)$.
- ↪ Algorytm Naive Bayes to prosta metoda wykorzystująca prawdopodobieństwo przynależności każdej z cech wzorca do klasy.
- ↪ Upraszcza on obliczenia zakładając, że wpływ różnych cech na prawdopodobieństwa przynależności do klas jest niezależny. **To właśnie jego naiwność.**

Naiwny klasyfikator bayesowski (ii)

- ↪ Prawdopodobieństwo przynależności do klasy na podstawie wartości cechy nazywamy **prawdopodobieństwem warunkowym**.

Naiwny klasyfikator bayesowski (ii)

- ↪ Prawdopodobieństwo przynależności do klasy na podstawie wartości cechy nazywamy **prawdopodobieństwem warunkowym**.
- ↪ Iloczyn prawdopodobieństw warunkowych po cechach określa prawdopodobieństwo przynależności do klasy całego wzorca.

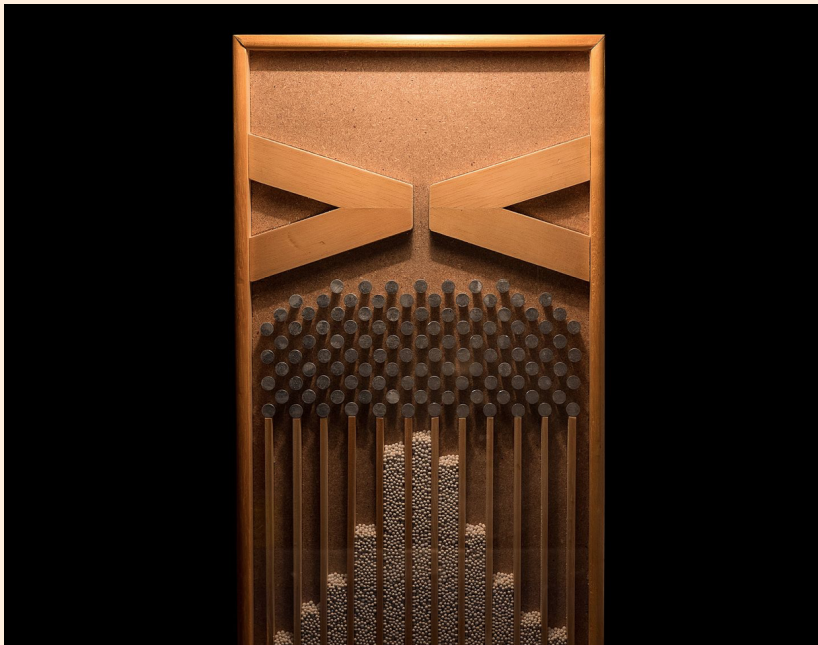
Naiwny klasyfikator bayesowski (ii)

- ↪ Prawdopodobieństwo przynależności do klasy na podstawie wartości cechy nazywamy **prawdopodobieństwem warunkowym**.
- ↪ Iloczyn prawdopodobieństw warunkowych po cechach określa prawdopodobieństwo przynależności do klasy całego wzorca.
- ↪ Najczęściej wyjaśniamy go na przykładzie cech jakościowych, ale bardziej przydatna jest jego ilościowa wersja.

Teleturnieje takie mądre



Deska Galtona



Rozkład normalny

Funkcja gęstości prawdopodobieństwa

$$\varphi(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\Pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Rozkład normalny

Funkcja gęstości prawdopodobieństwa

$$\varphi(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\Pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

↗ x — zmienna objaśniana

Rozkład normalny

Funkcja gęstości prawdopodobieństwa

$$\varphi(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\Pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

↗ x — zmienna objaśniana

↗ μ — wartość oczekiwana

Rozkład normalny

Funkcja gęstości prawdopodobieństwa

$$\varphi(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\Pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- ↪ x — zmienna objaśniana
- ↪ μ — wartość oczekiwana
- ↪ σ — odchylenie standardowe

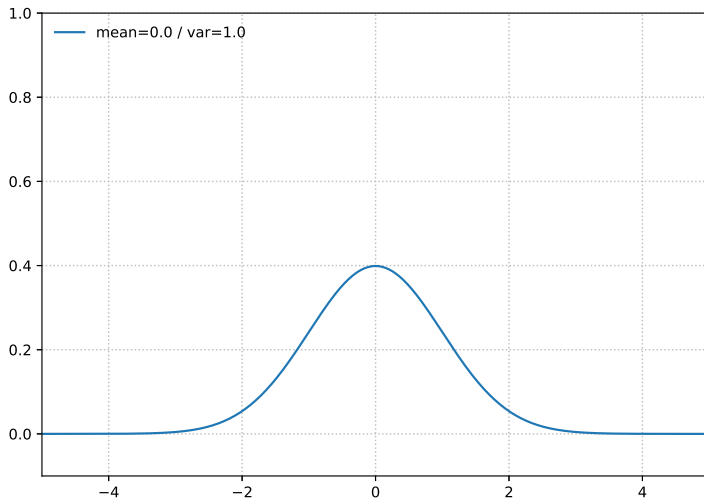
Rozkład normalny

Funkcja gęstości prawdopodobieństwa

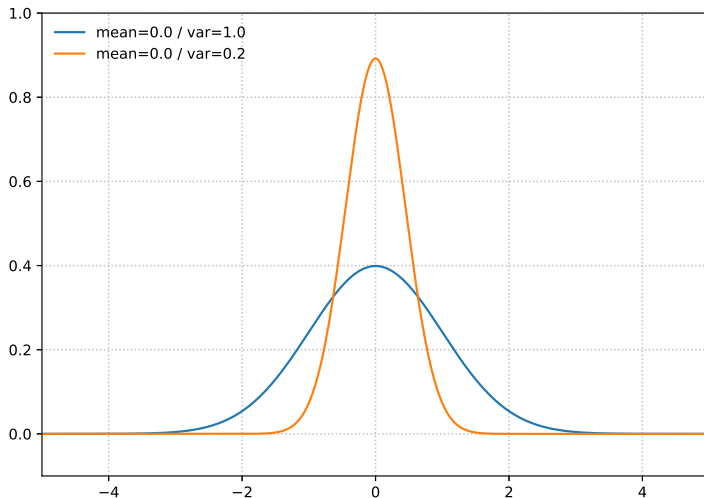
$$\varphi(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\Pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- ↪ x — zmienna objaśniana
- ↪ μ — wartość oczekiwana
- ↪ σ — odchylenie standardowe
- ↪ σ^2 — wariancja

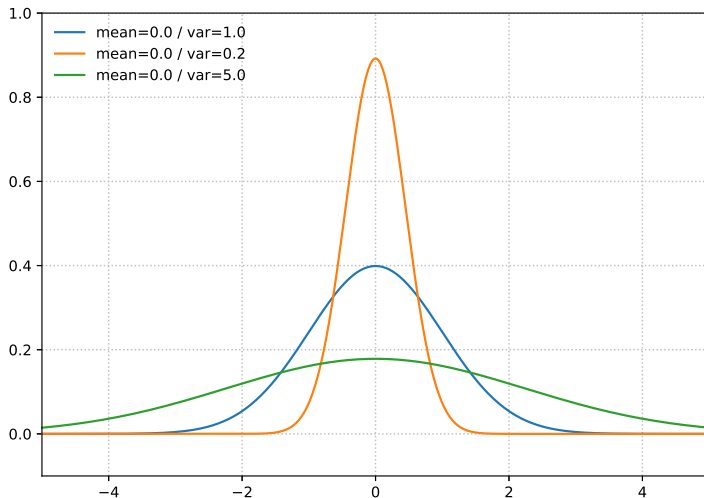
Rozkład normalny



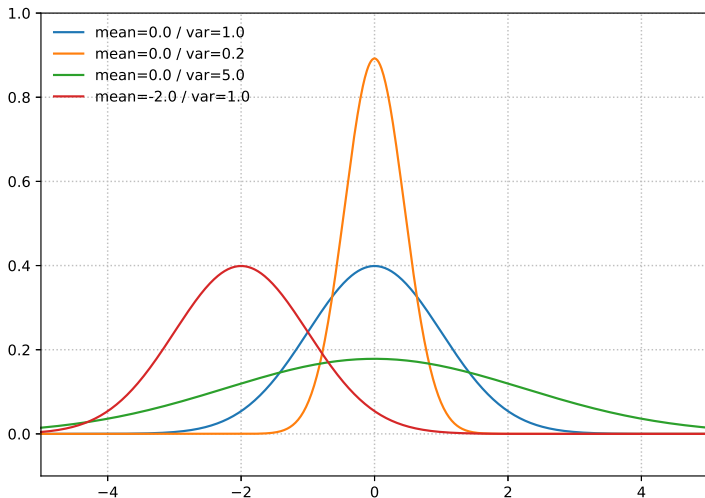
Rozkład normalny



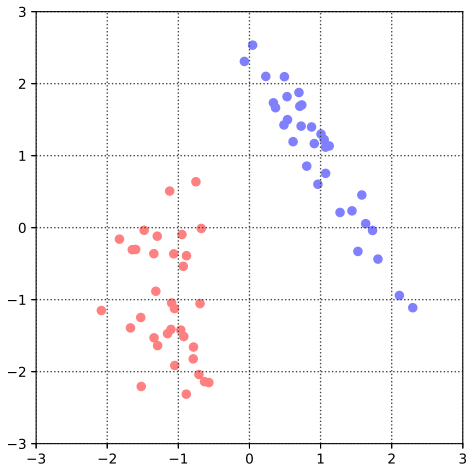
Rozkład normalny



Rozkład normalny

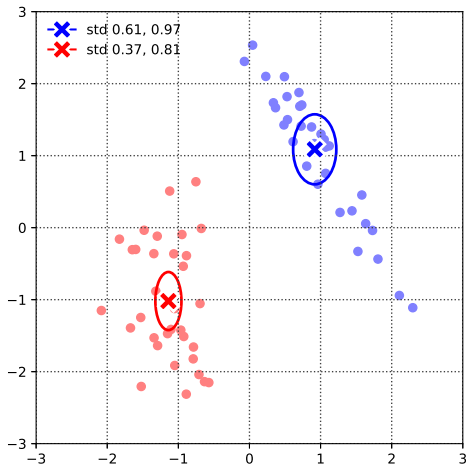


Naiwny Bayes



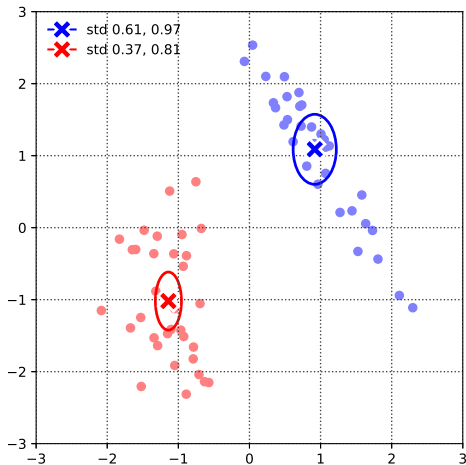
⇒ Dysponujemy zbiorem uczącym składającym się z dwóch cech ilościowych.

Naiwny Bayes



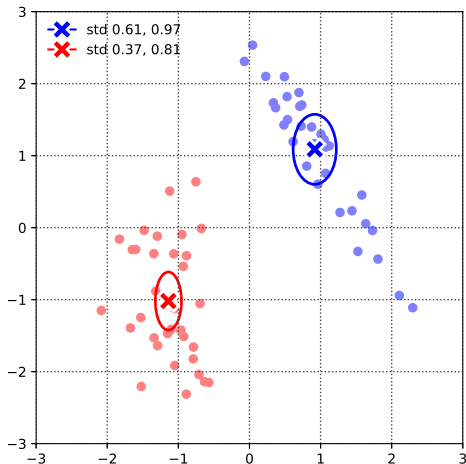
→ Dla **każdej z klas** problemu wyznaczamy statystycznie jej:

Naiwny Bayes



⇒ Dla **każdej z klas** problemu wyznaczamy statystycznie jej:
× centroid,

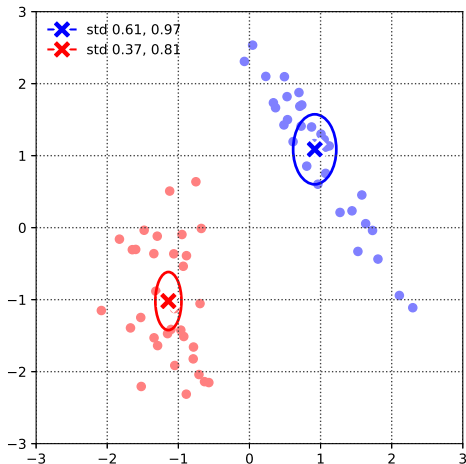
Naiwny Bayes



→ Dla **każdej z klas** problemu wyznaczamy statystycznie jej:

- × centroid,
- odchylenie standardowe przynależących do niej wzorców.

Naiwny Bayes

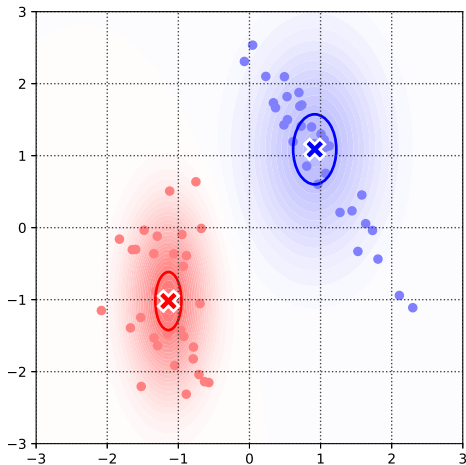


→ Dla **każdej z klas** problemu wyznaczamy statystycznie jej:

- × centroid,
- odchylenie standardowe przynależących do niej wzorców.

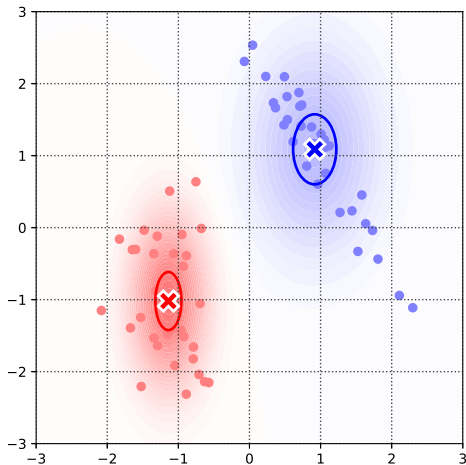
→ Parametry wyliczamy **osobno dla każdej cechy**.

Naiwny Bayes



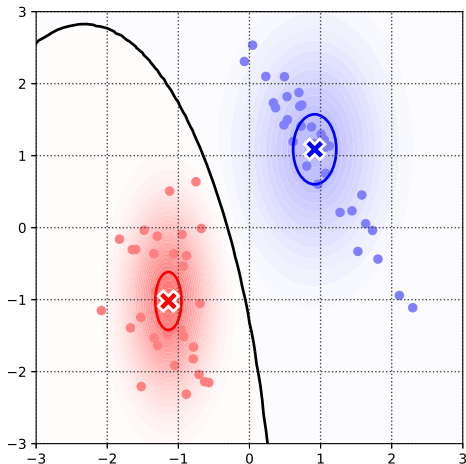
⇒ Na każdy centroid klasy nakładamy funkcję potencjałową zgodną z rozkładem normalnym.

Naiwny Bayes



- Na każdy centroid klasy nakładamy funkcję potencjałową zgodną z rozkładem normalnym.
- Jak bardzo spójna z prawdziwymi danymi jest to konstrukcja?

Naiwny Bayes



⇒ Granica decyzyjna przebiega w punkcie, w którym potencjały prawdopodobieństwa się równoważą.

~~~~~

Co daje nam naiwność?

~~~~~