DeepDive™

Po co nam redukcja wymiarów?

DeepDive™

Redukcja wymiarów

Wizualizacja danych w 2D/3D

Przyspieszenie obliczeń

Usuwanie szumu

Eliminacja redundancji

Kompresja danych

Lepsze zrozumienie struktury danych i ich wzorców

Klątwa wielowymiarowości

DeepDive™

Klątwa wielowymiarowości

Wraz ze wzrostem liczby wymiarów:

dane stają się coraz rzadsze,

odległości między punktami tracą sens,

modele uczą się wolniej i łatwiej przeuczyć model,
rośnie koszt obliczeniowy.

Metody liniowe

PCA
(Principal Component Analysis)

najpopularniejsza, przekształca dane w nowe, ortogonalne osie maksymalnej wariancji

LDA
(Linear Discriminant Analysis)

uwzględnia przynależność do klas optymalna dla klasyfikacji

ICA
(Independent Component Analysis)

szuka komponentów statystycznie niezależnych

Metody nieliniowe

t-SNE

(t-distributed Stochastic Neighbor Embedding)

świetna do wizualizacji, zachowuje lokalną strukturę danych

UMAP

(Uniform Manifold Approximation and Projection)

szybsze od t-SNE, zachowuje więcej globalnej struktury

Isomap

oparta na geodezyjnych odległościach na kolektorze danych

Metody oparte o uczenie maszynowe

Autoenkodery (**Autoencoders**)
sieci neuronowe uczące się kompresji danych

Feature selection (np. LASSO, SelectKBest)
wybierają najważniejsze cechy bez tworzenia nowych

DeepDive™

Jak działa LDA?

Liniowa analiza dyskryminacyjna

metoda uczenia nadzorowanego, która znajduje takie kombinacje cech, które najlepiej oddzielają od siebie różne klasy. Jej celem jest zmniejszenie liczby wymiarów, zachowując przy tym maksymalną ilość informacji potrzebną do rozróżnienia kategorii.
Uzyskane nowe cechy mogą być wykorzystane do budowy klasyfikatora lub jako wstępny krok w analizie danych.

super do klasyfikacji i wizualizacji,
jeśli mamy dane z etykietami

zakłada, że dane w klasach
są normalnie rozłożone i mają równą kowariancję

w odróżnieniu od PCA,
LDA maksymalizuje separację klas, a nie wariancję ogólną

DeepDive™

CEL

Znalezienie kombinacji liniowych cech, które najlepiej odróżniają klasy w danych.

1. **Obliczenie średnich klasowych**

   Dla każdej klasy obliczamy wektor średnich wartości cech

2. **Obliczenie macierzy rozrzutu wewnątrzklasowego**

   Reprezentuje wariancję cech w obrębie każdej klasy

3. **Obliczenie macierzy rozrzutu międzyklasowego**

   Reprezentuje wariancję cech między średnimi wartościami cech różnych klas.

4. **Rozwiązanie problemu wartości własnych**

   Pozwala znaleźć kierunki w przestrzeni cech, które maksymalizują separację między klasami

5. **Wybór głównych kierunków dyskryminacji**

   Wybieramy k-1 wektorów własnych odpowiadających największym wartościom własnym

6. **Transformacja danych**

   Rzutujemy oryginalne dane na wybrane kierunki, uzyskując nową przestrzeń o mniejszej liczbie wymiarów

Rozpoznawanie płci na podstawie wzrostu i masy ciała

Osoba	Wzrost (cm)	Masa (kg)	Płeć
A	160	50	Kobieta (0)
B	165	55	Kobieta (0)
C	170	54	Kobieta (0)
D	180	75	Mężczyzna (1)
E	175	70	Mężczyzna (1)
F	185	85	Mężczyzna (1)

DeepDive™

1

Obliczenie średnich klasowych

Dla każdej klasy obliczamy wektor średnich wartości cech

Kobiety (klasa 0):

$$\mu_0 = \left[\frac{160 + 165 + 170}{3}, \frac{50 + 55 + 54}{3}\right] = [165, 53]$$

Mężczyźni (klasa 1):

$$\mu_1 = \left[\frac{180 + 175 + 185}{3}, \frac{75 + 70 + 85}{3}\right] = [180, 76.7]$$

Średnia globalna:

$$\mu = \left[\frac{160 + 165 + 170 + 180 + 175 + 185}{6}, \frac{50 + 55 + 54 + 75 + 70 + 85}{6}\right] = [172.5, 66.5]$$

DeepDive™

2 Obliczenie macierzy rozrzutu wewnątrzklasowego

Reprezentuje wariancję cech w obrębie każdej klasy

Dla klasy 0 (kobiety):

Próbka	$x_i - \mu_0$	$(x_i - \mu_0)(x_i - \mu_0)^T$
A	[-5, -3]	$\begin{bmatrix} 25 & 15 \\ 15 & 9 \end{bmatrix}$
B	[0, 2]	$\begin{bmatrix} 0 & 0 \\ 0 & 4 \end{bmatrix}$
C	[5, 1]	$\begin{bmatrix} 25 & 5 \\ 5 & 1 \end{bmatrix}$

$$S_{W0} = \sum = \begin{bmatrix} 50 & 20 \\ 20 & 14 \end{bmatrix}$$

DeepDive™

2 Obliczenie macierzy rozrzutu wewnątrzklasowego

Reprezentuje wariancję cech w obrębie każdej klasy

Dla klasy 1 (mężczyźni):

Próbka	$x_i - \mu_1$	$(x_i - \mu_1)(x_i - \mu_1)^T$
D	[0, -1.7]	$\begin{bmatrix} 0 & 0 \\ 0 & 2.89 \end{bmatrix}$
E	[-5, -6.7]	$\begin{bmatrix} 25 & 33.5 \\ 33.5 & 44.89 \end{bmatrix}$
F	[5, 8.3]	$\begin{bmatrix} 25 & 41.5 \\ 41.5 & 68.89 \end{bmatrix}$

$$S_{W1} = \sum = \begin{bmatrix} 50 & 75 \\ 75 & 116.67 \end{bmatrix}$$

Całkowita macierz S_W:

$$S_W = S_{W0} + S_{W1} = \begin{bmatrix} 100 & 95 \\ 95 & 130.67 \end{bmatrix}$$

3

Obliczenie macierzy rozrzutu międzyklasowego

Reprezentuje wariancję cech między średnimi wartościami cech różnych klas.

$$S_B = \sum n_k(\mu_k - \mu)(\mu_k - \mu)^T$$

Dla klasy 0:

$$\mu_0 - \mu = [165 - 172.5, 53 - 66.5] = [-7.5, -13.5] \Rightarrow 3 \cdot [-7.5, -13.5]^T[-7.5, -13.5] = 3 \cdot \begin{bmatrix} 56.25 & 101.25 \\ 101.25 & 182.25 \end{bmatrix}$$

Dla klasy 1:

$$\mu_1 - \mu = [180 - 172.5, 76.7 - 66.5] = [7.5, 10.2] \Rightarrow 3 \cdot \begin{bmatrix} 56.25 & 76.5 \\ 76.5 & 104.04 \end{bmatrix}$$

Suma:

$$S_B = \begin{bmatrix} 337.5 & 532.5 \\ 532.5 & 858.87 \end{bmatrix}$$

DeepDive™

4

Rozwiązanie problemu wartości własnych

Pozwala znaleźć kierunki w przestrzeni cech, które maksymalizują separację między klasami

Obliczamy:

$$S_W^{-1} S_B$$

Znajdujemy największą wartość własną i jej wektor,
to będzie kierunek LDA
(przy małej liczbie cech: korzystamy z numpy
lub rysujemy geometrycznie).

5

Wybór głównych kierunków dyskryminacji

Wybieramy k-1 wektorów własnych odpowiadających największym wartościom własnym

$$k-1 = 2-1 = 1$$

DeepDive™

6

Transformacja danych

Rzutujemy oryginalne dane na wybrane kierunki, uzyskując nową przestrzeń o mniejszej liczbie wymiarów

Każdy punkt:

$$z_i = w^T x_i$$

daje jedną wartość liczbową = współrzędna na osi LDA (1D).

Case study

Zbiór danych Wine z biblioteki scikit-learn.

Liczba klas: 3 (trzy różne odmiany wina)
Liczba cech: 13 (chemiczne właściwości win)
Liczba próbek: 178

Przegląd danych

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	color_intensity
0	14.23	1.71	2.43	15.6	127.0	2.80	3.06	0.28	2.29	5.64
1	13.20	1.78	2.14	11.2	100.0	2.65	2.76	0.26	1.28	4.38
2	13.16	2.36	2.67	18.6	101.0	2.80	3.24	0.30	2.81	5.68
3	14.37	1.95	2.50	16.8	113.0	3.85	3.49	0.24	2.18	7.80
4	13.24	2.59	2.87	21.0	118.0	2.80	2.69	0.39	1.82	4.32
...	...	...	...	...	...	...	...	...	...	...
173	13.71	5.65	2.45	20.5	95.0	1.68	0.61	0.52	1.06	7.70
174	13.40	3.91	2.48	23.0	102.0	1.80	0.75	0.43	1.41	7.30
175	13.27	4.28	2.26	20.0	120.0	1.59	0.69	0.43	1.35	10.20
176	13.17	2.59	2.37	20.0	120.0	1.65	0.68	0.53	1.46	9.30
177	14.13	4.10	2.74	24.5	96.0	2.05	0.76	0.56	1.35	9.20

178 rows × 14 columns

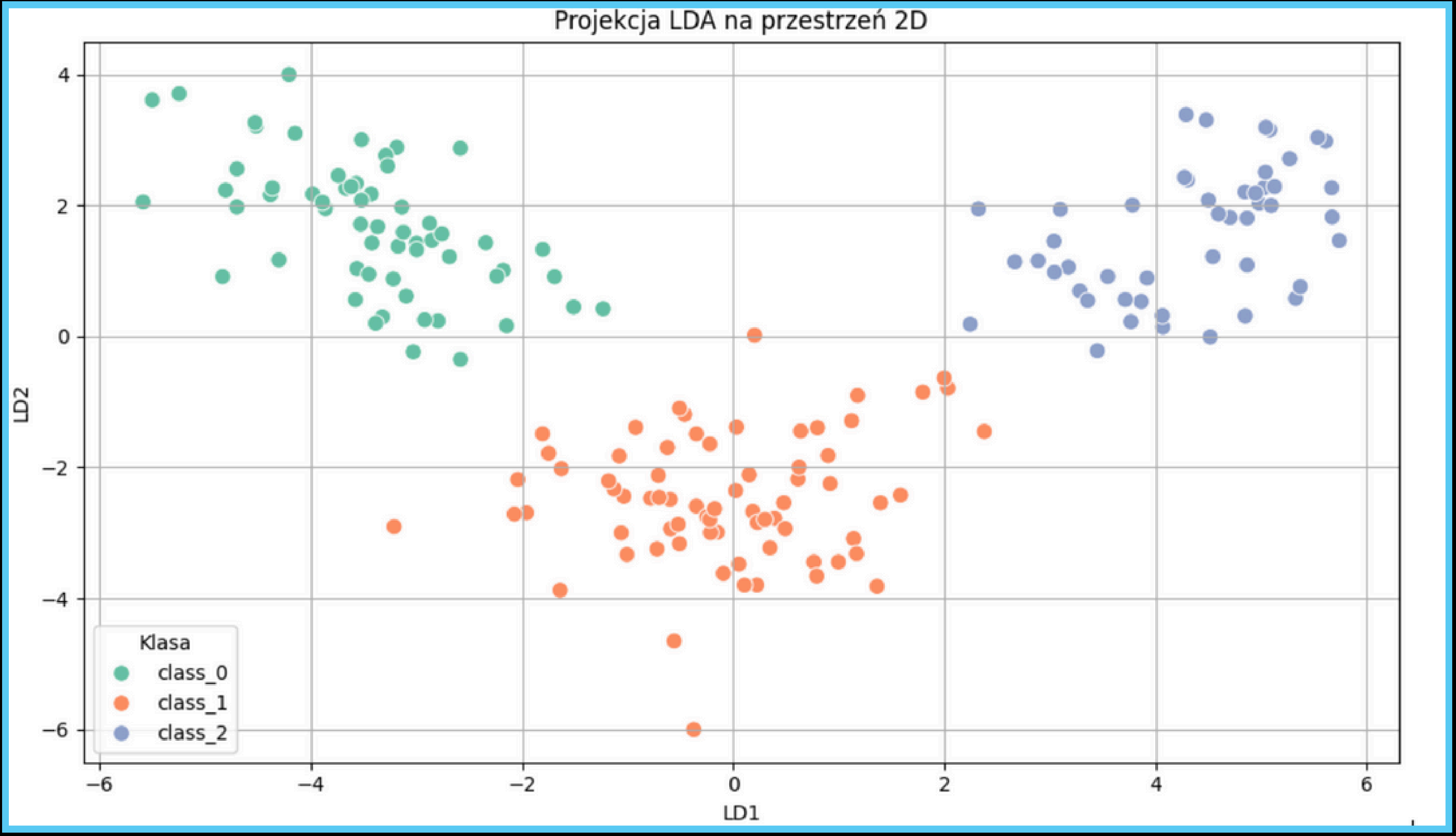
DeepDive™

Dane transformowane

	LD1	LD2	class	class_name
0	-4.700244	1.979138	0	class_0
1	-4.301958	1.170413	0	class_0
2	-3.420720	1.429101	0	class_0
3	-4.205754	4.002871	0	class_0
4	-1.509982	0.451224	0	class_0
...	...	...	...	...
173	4.291508	3.390332	2	class_2
174	4.503296	2.083546	2	class_2
175	5.047470	3.196231	2	class_2
176	4.276155	2.431388	2	class_2
177	5.538086	3.042057	2	class_2

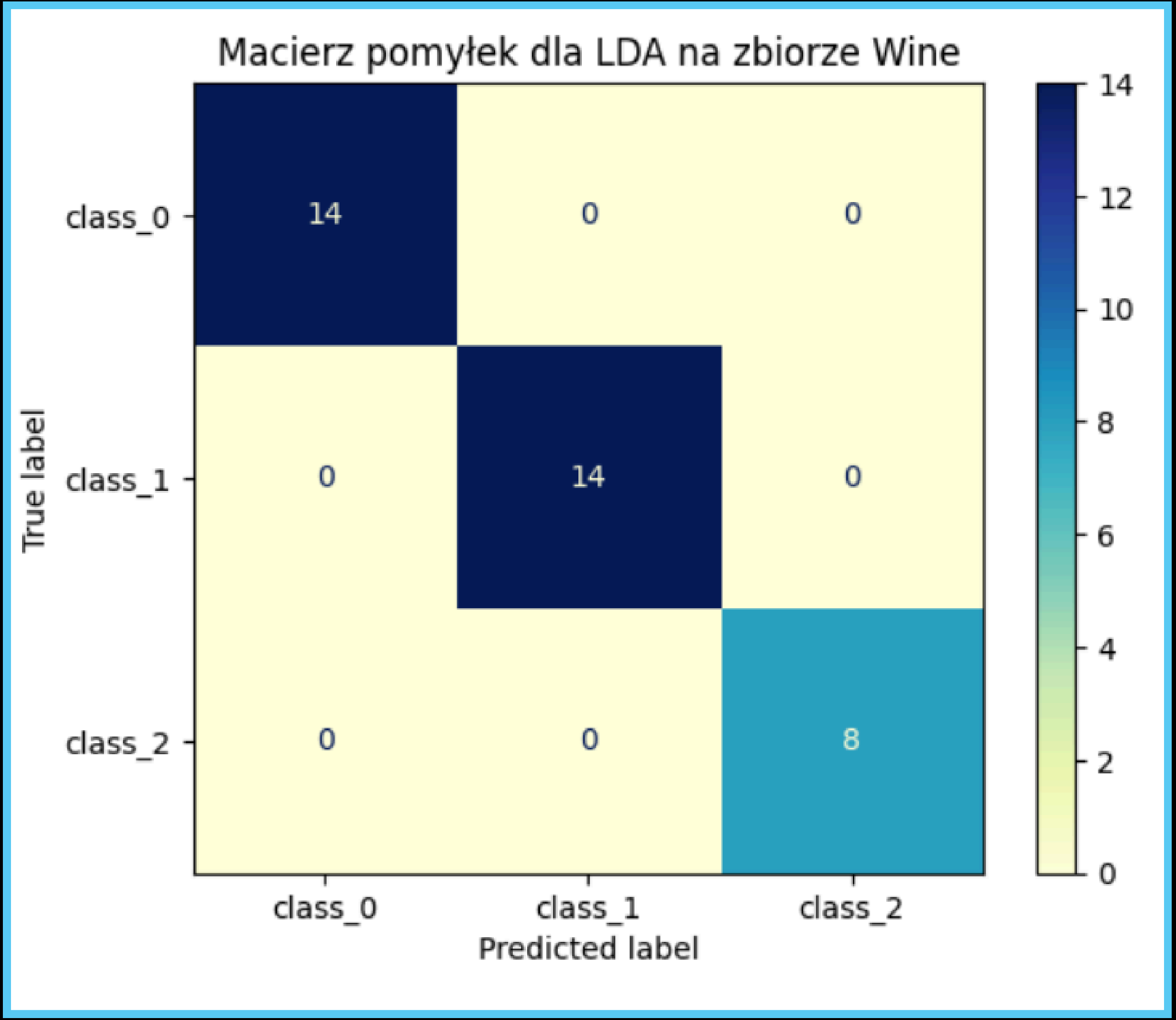
178 rows × 4 columns

Projekcja danych



DeepDive™

Predykcja



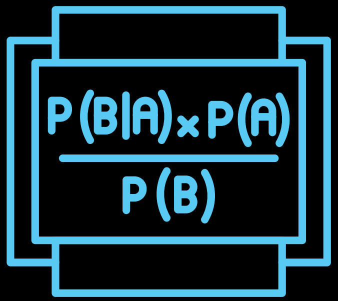
Średnia dokładność (CV): 96.63%

LDA to nie tylko redukcja – to klasyfikator Bayesowski

LDA zakłada, że dane w każdej klasie pochodzą z **wielowymiarowego rozkładu normalnego**.

Zakłada też **wspólną macierz kowariancji** dla wszystkich klas.

Bazując na tym, LDA dokonuje klasyfikacji Bayesowskiej, wybierając klasę z **najwyższym prawdopodobieństwem.**



DeepDive™

LDA to nie tylko redukcja – to klasyfikator Bayesowski

Argumenty funkcji LDA ze scikit-learna

solver
Algorytm: 'svd', 'lsqr', 'eigen'

shrinkage
Regularizacja: 'auto', float, 'none'

n_components
Ile wymiarów LDA chcesz uzyskać (max = liczba klas – 1)

priors
Ręczne ustawienie prawdopodobieństw klas

store_covariance
Czy przechować macierz kowariancji

tol
Tolerancja numeryczna dla eigen solvera

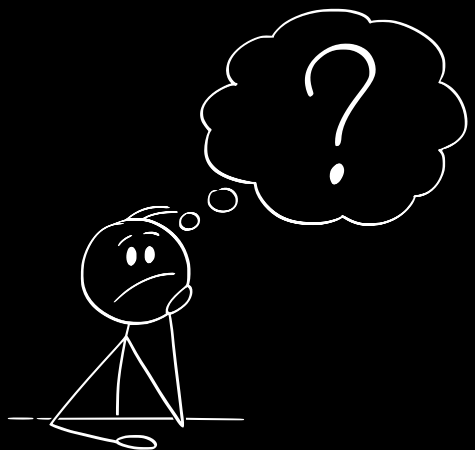
DeepDive™

Jak wybrać solver?

svd - szybki, bez kowariancji, bez shrinkage (najczęściej używany)

lsqr - obsługuje shrinkage, dobry do wysokowymiarowych danych

eigen - jak lsqr, ale używa innej metody własnej, wolniejszy



DeepDive™

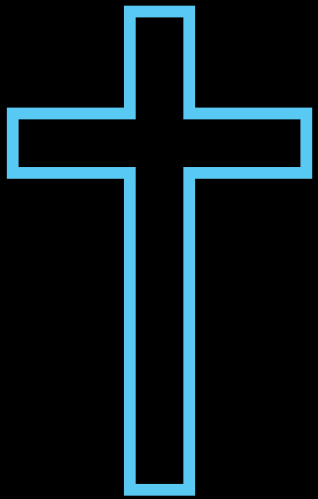
Shrinkage

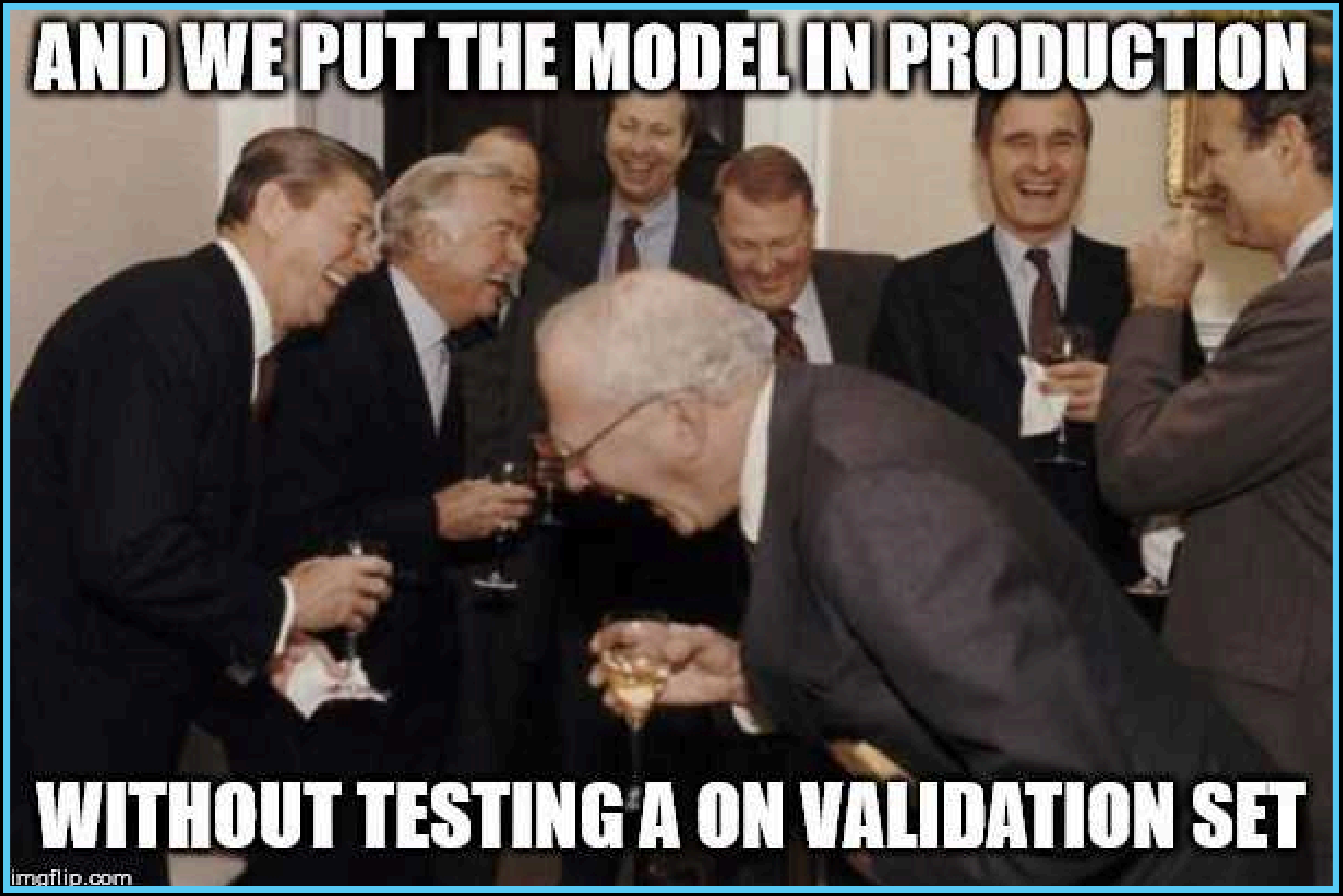
Gdy liczba cech jest wysoka względem liczby próbek, macierz kowariancji może być niestabilna. Shrinkage uśrednia ją z macierzą jednostkową.

Pomaga uniknąć nadmiernego dopasowania (overfittingu).

Walidacja krzyżowa

Pomaga oszacować, jak dobrze LDA radzi sobie z danymi nieznanymi.




AND WE PUT THE MODEL IN PRODUCTION
WITHOUT TESTING A ON VALIDATION SET
imgflip.com

DeepDive™

ZALETY I OGRANICZENIA LDA

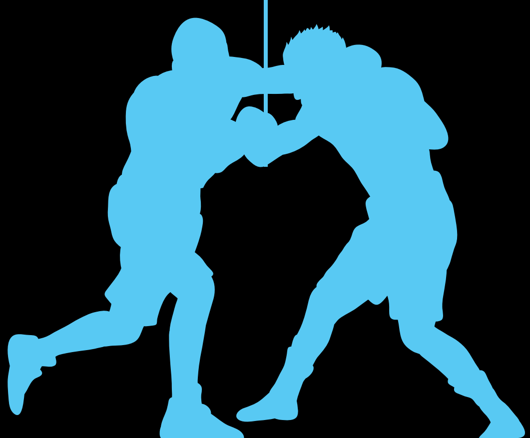
- Efektywna redukcja wymiarowości przy zachowaniu informacji o klasach.
- Poprawa wydajności algorytmów klasyfikacyjnych.
- Lepsza interpretowalność danych dzięki projekcji na mniejszą liczbę wymiarów.

- Założenie liniowej separowalności klas, co może nie być spełnione w rzeczywistości.
- Wrażliwość na obecność wartości odstających.
- Wymaga, aby liczba próbek w każdej klasie była większa niż liczba cech.

DeepDive™

LDA vs PCA

metoda uczenia z nadzorem	metoda uczenia bez nadzoru
uwzględnia etykiety klas	nie uwzględnia etykiet klas
dąży do maksymalizacji separacji między klasami	koncentruje się na maksymalnej wariancji danych



DeepDive™

DeepDive™