Metody Eksploracji Danych (projekt)
Opracowanie modelu regresji

Jakub Matłacz, Kamil Trysiński

23 stycznia 2022

Spis treści

1 Przygotowanie danych

Po wstępnym przetworzeniu i analizie eksploracyjnej - wiele cech jest binarnych. Aby zrównać wpływ wszystkich tych cech - zostały one znormalizowane do zakresu. Wybraliśmy zakres $\langle 1, 2\rangle$ - pozwala on na uniknięcie problemów z małymi liczbami czy dzieleniem przez 0 w metrykach. Znormalizowaliśmy także zmienną opisywaną - testy wykazały, że brak skalowania dla tej cechy powoduje gorsze wyniki klasyfikacji.

2 Dummy regressor

Przed rozpoczęciem eksperymentów stworzyliśmy model atrapę, który dokonuje regresji przy użyciu prostych strategii - przykładowo dla danych ze zbioru treningowego zwraca wynik stanowiący jego średnią lub medianę. Stworzyliśmy kilka prostych strategii i sprawdziliśmy wyniki testowania. Okazało się, że wyniki są nie mniejsze niż około 12% MAPE. Wniosek jest więc następujący - wybrany końcowo model powinien osiągać wynik znacznie lepszy od 12%.

3 Testowanie modeli

Dla każdego z modeli chcieliśmy dobrać jak najlepsze hiperparametry. Wyłoniliśmy je w procesie walidacji krzyżowej z 5-krotnym podziałem. Każdorazowo wyniki były przedstawiane na wykresie zależności wyników od hiperparametru, wraz z odchyleniami standardowymi.

Aby uzyskać jak najlepsze wyniki wyboru - szukaliśmy jednego parametru na raz, zakładając a priori jakieś wartości innych parametrów. Np. jeśli chcemy dla modelu wybrać parametr epsilon i współczynnik regularyzacji lambda - najpierw założymy jakieś lambda zmieniając kolejno wartości epsilon, po czym dla najlepszego epsilon - będziemy zmieniać kolejno wartości lambda. Po przeprowadzeniu takich testów - powinniśmy uzyskać całkiem dobre wartości hiperparametrów.

W tym celu używaliśmy funkcji *plot_validation_curve* - dostępna jest ona w naszej bibliotece pomocniczej, załączonej wraz ze sprawozdaniem, a stworzona została na podstawie propozycji ze strony scikit_learn. Wynik, lub "score", stanowił "neg_mean_absolute_error" - czyli ujemna wartość absolutna błędu regresji. Dzięki iloczynowi z "-1" błąd został zamieniony w wynik, który można maksymalizować zamiast minimalizować. Poprawia to łatwość interpretacji i lepiej wpisuje się w konwencję.

Narysowana zostały również trzy inne wykresy. Pierwszym z nich jest krzywa uczenia dla każdego z modeli. Pokazuje ona wyniki walidacji krzyżowej dla różnych wielkości zbioru treningowego. Pozwala to wyciągnąć wnioski odnośnie dostateczności rozmiaru zbioru. Określić można czy jego zwiększenie miałoby szansę polepszyć wyniki.

Kolejny załączony wykres przedstawia skalowalność każdego z modeli. Pokazuje on zależność czasu dopasowania od wielkości zbioru dopasowywanego.

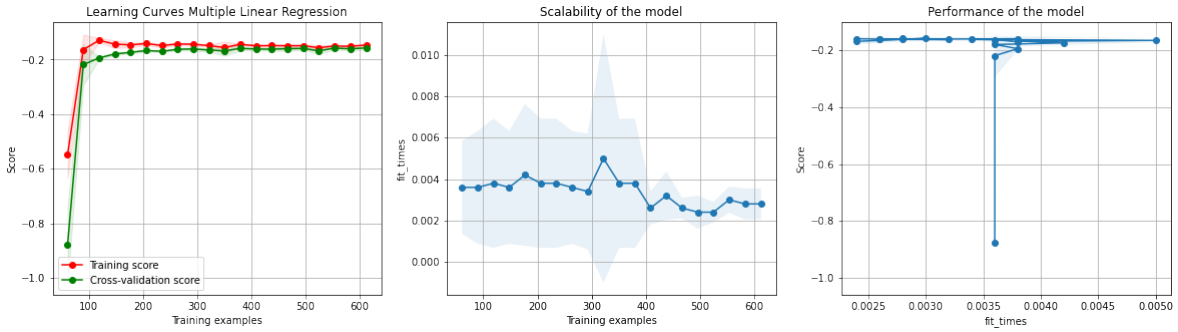
Ostatni wykres to efektywność modelu, czyli zależność wyniku od czasu dopasowania - pokazuje jak czas dopasowania przekłada się na wynik trenowania, czyli jakość dopasowania.

Na samym końcu testu modelu, po wybraniu hiperparametrów i zaakceptowaniu krzywej uczenia, wywołana zostaje procedura testowania modelu na zbiorze testowym, którego nigdy wcześniej model "nie widział". Wynikiem jest, łatwy w interpretacji, mean absolute percentage error (MAPE) czyli średni absolutny błąd procentowy. Pozwala on nam na zrozumienie jak bardzo model myli się dokonując regresji na nowych danych.

4 Przetestowane modele

4.1 Wieloraka regresja liniowa

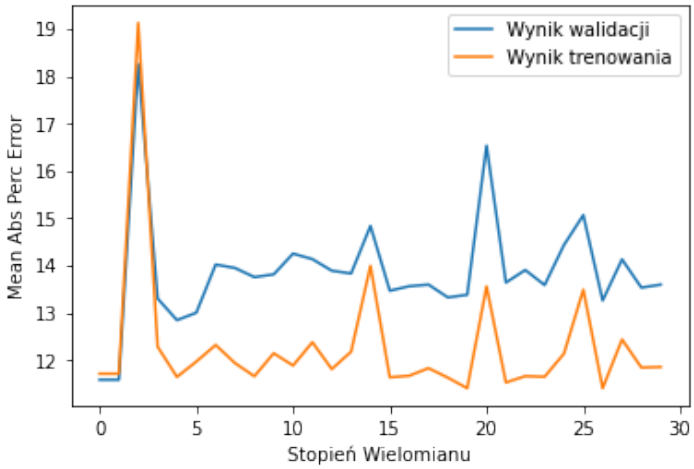
Regresja bez żadnych hiperparametrów. Błąd testowania na poziomie 11.6%, a uczenia na poziomie 11.7% - jest więc nieznacznie lepszy niż najprostsze dummy podejścia. Krzywa uczenia pokazuje nam, że mamy wystarczająco danych i dodanie nowych nie poprawi nam już znacząco dokładności modelu. Model jest także dosyć skalowalny, ale znowu - nie poprawia to jakości predykcji.



Rysunek 1: Krzywe uczenia, skalowalność modelu

4.2 Wieloraka regresja wielomianowa

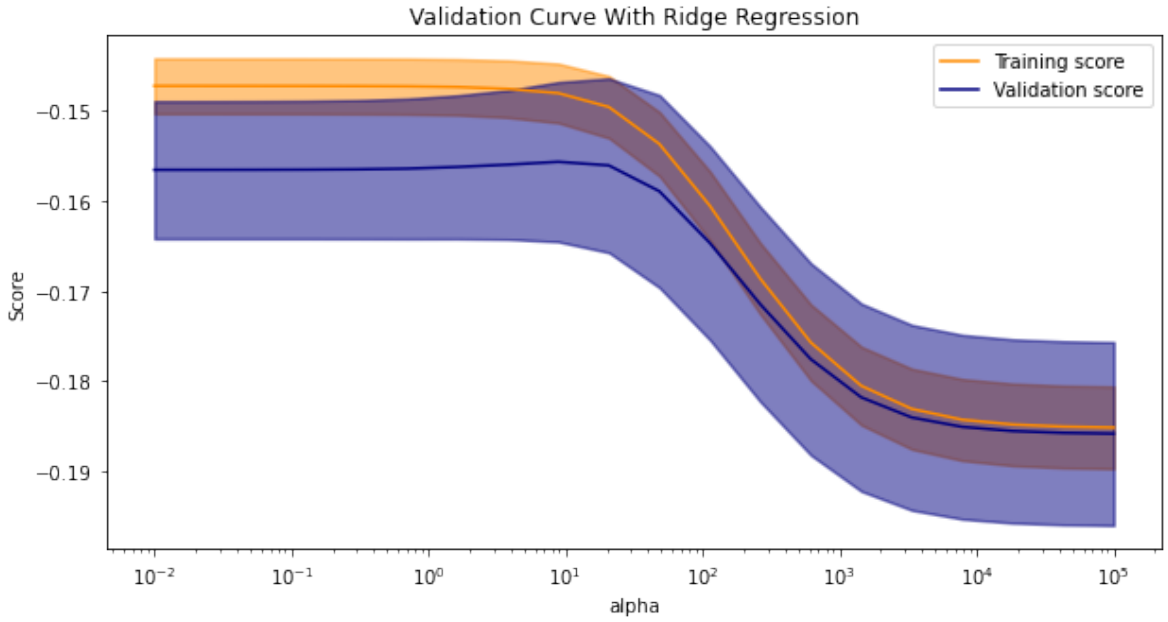
Testowana dla różnych stopni wielomianu. Najlepsze wyniki osiągał stopień 0, ale znowu - wynik testowania na poziomie 11.7%, a uczenia na poziomie 11.6%. Jest to więc wynik nieznacznie lepszy niż nasz dummy model, na pewno można znaleźć lepszy model.



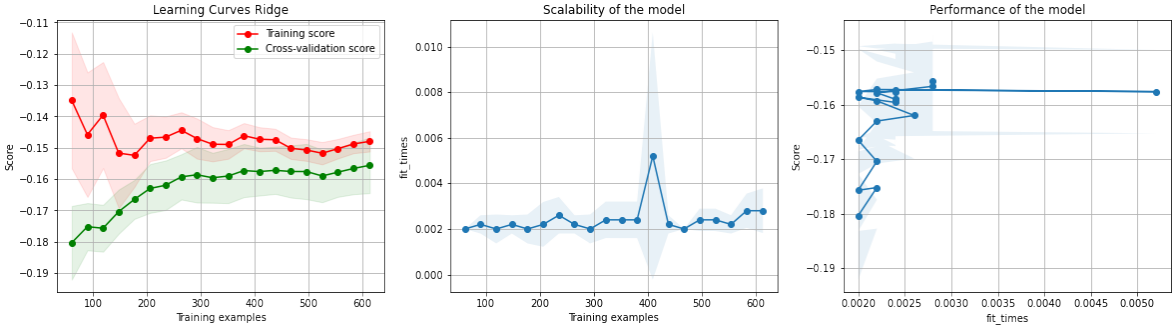
Rysunek 2: Zależność błędu od stopnia wielomianu

4.3 Regresja grzbietowa

Najpierw należało wyznaczyć współczynnik regularyzacji alpha. Testowane wartości zawierały się w zakresie od 10^{-2} do 10^{5}. Najlepszym parametrem okazała się wartość alpha równa ok. 8.86. Dla wyznaczonej wartości współczynnika regularyzacji - błąd testowania wyniósł ok. 11.9%, a błąd uczenia również ok. 11.9%. Na wykresach widzimy także, że model nie zyska już wiele na zwiększeniu ilości danych i jest dosyć skalowalny. Biorąc pod uwagę wszystkie te dane - model grzbietowy jest porównywalny z naszymi dummy.



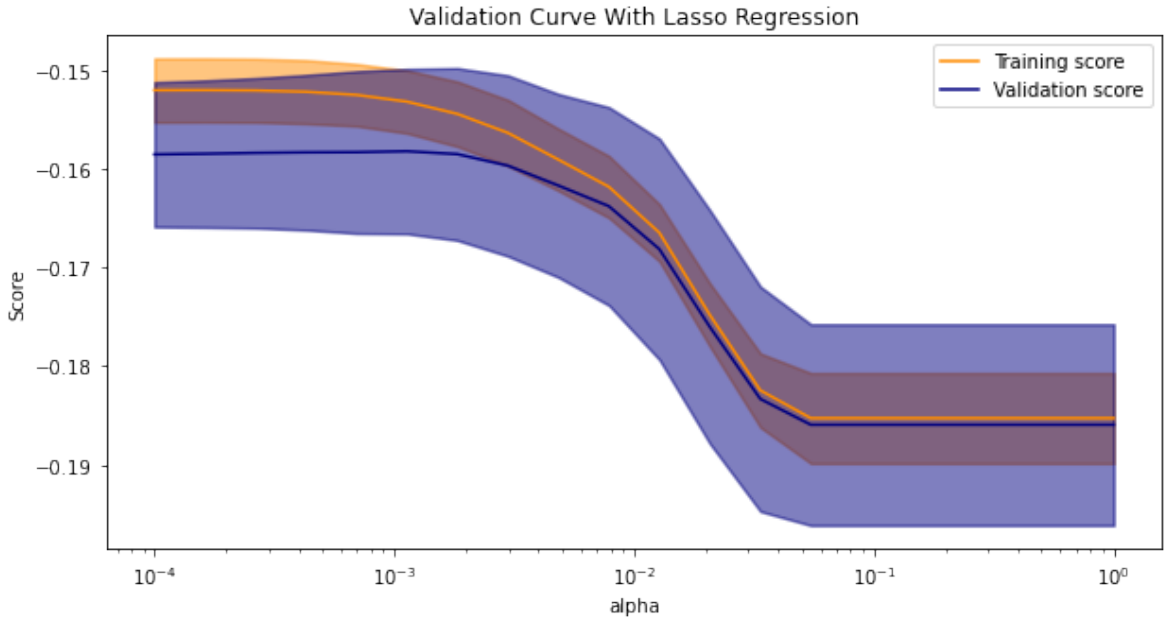
Rysunek 3: Wyniki w zależności od alpha



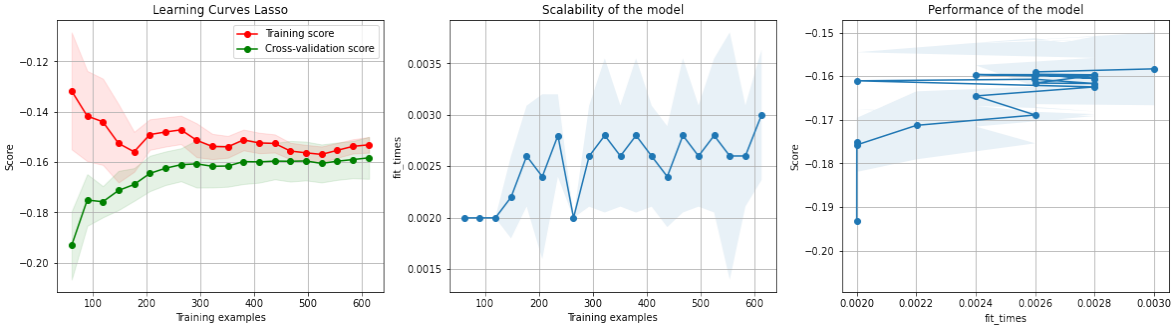
Rysunek 4: Krzywe uczenia, skalowalność modelu

4.4 Regresja Lasso

Podobnie jak w przypadku regresji grzbietowej - należało najpierw wyznaczyć parametr alpha. Tym razem - testowane wartości zawierały się w zakresie od 10^{-4} do 10^{0}. Najlepszym parametrem okazała się wartość alpha równa ok. 0.0011. Dla wyznaczonej wartości współczynnika regularyzacji - błąd testowania wyniósł ok. 12.18%, a błąd uczenia ok. 12.35%. Poziom uczenia i skalowalności jest podobny jak w przypadku regresji liniowej, tylko tym razem - nasze błędy są wyższe niż przy użyciu dummy modeli. Oznacza to, że regresja Lasso zdecydowanie nie przyda się w naszym przypadku.



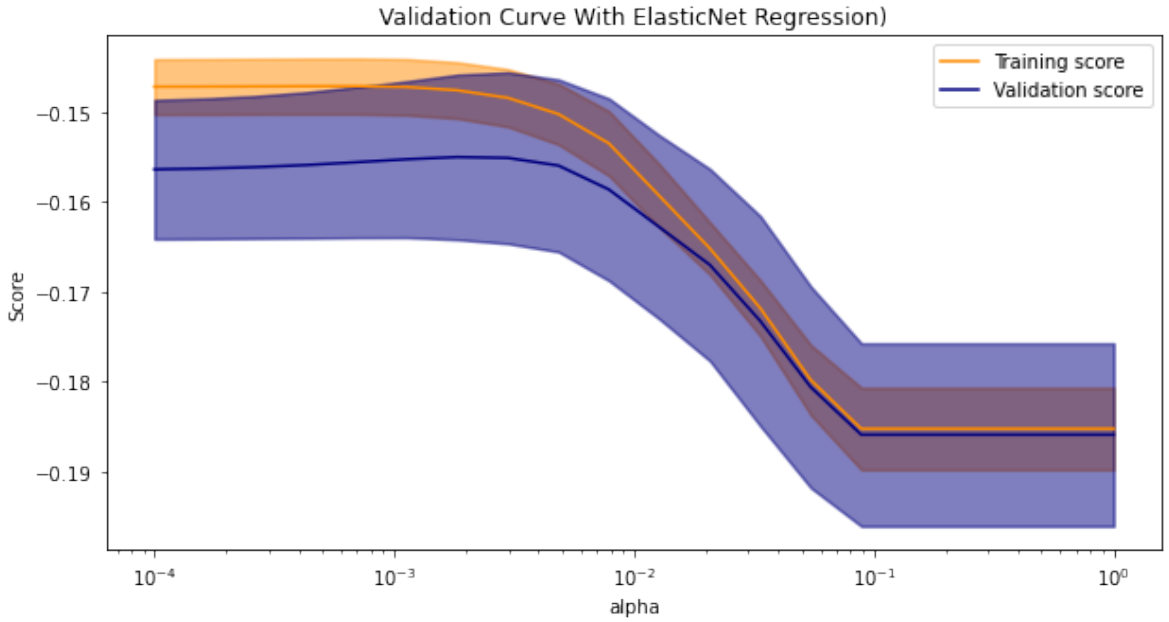
Rysunek 5: Wyniki w zależności od alpha



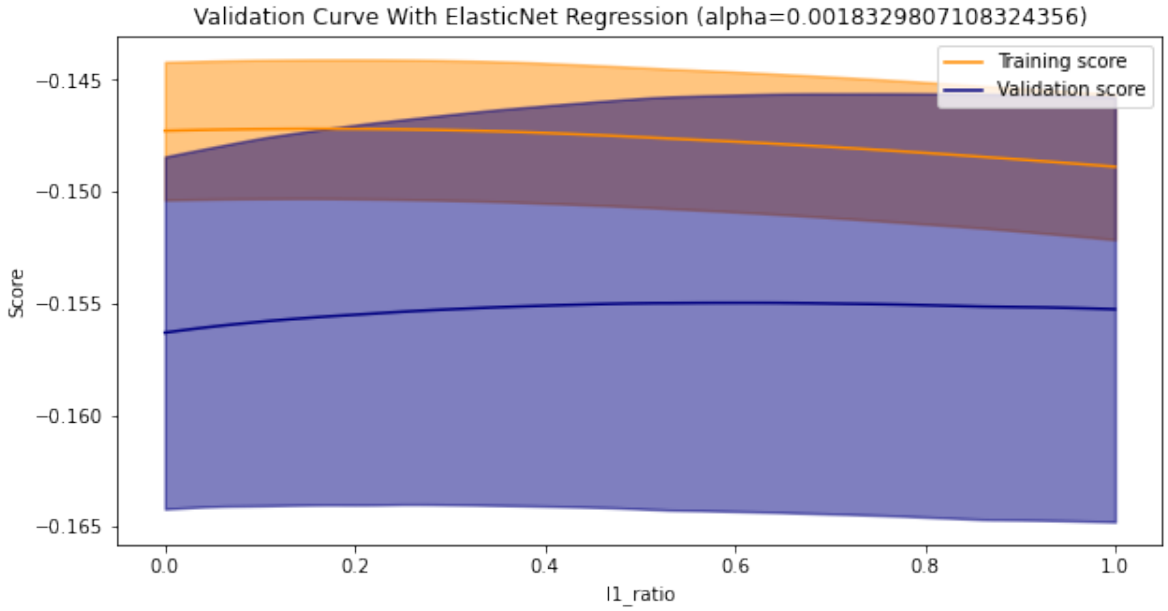
Rysunek 6: Krzywe uczenia, skalowalność modelu

4.5 ElasticNet

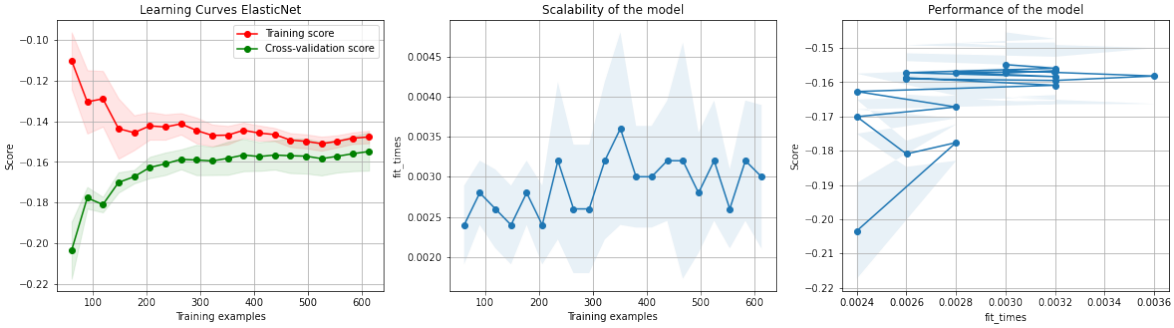
Regresja łącząca regularyzację L1 i L2. Tutaj musimy dokonać wyboru dwóch parametrów - współczynnika regularyzacji alpha i wpływu regularyzacji L1 (l1_ratio). Dla alpha - testowane wartości zawierały się w zakresie od 10^{-4} do 10^{0}, a dla l1_ratio - w zakresie od 0 do 1. Najlepsza wartość alpha wyniosła ok. 0.0018, a najlepsza wartość l1_ratio - 0.62. Dla wybranych parametrów błąd testowania wyniósł ok. 11.86%, a błąd uczenia ok. 11.91%. Podobnie jak poprzednie dwie regresje - model nie zyska na zwiększeniu zbioru danych i jest dosyć skalowalny, ale jego wyniki są porównywalne z modelami dummy.



Rysunek 7: Wyniki w zależności od alpha



Rysunek 8: Wyniki w zależności od l1_ratio



Rysunek 9: Krzywe uczenia, skalowalność modelu

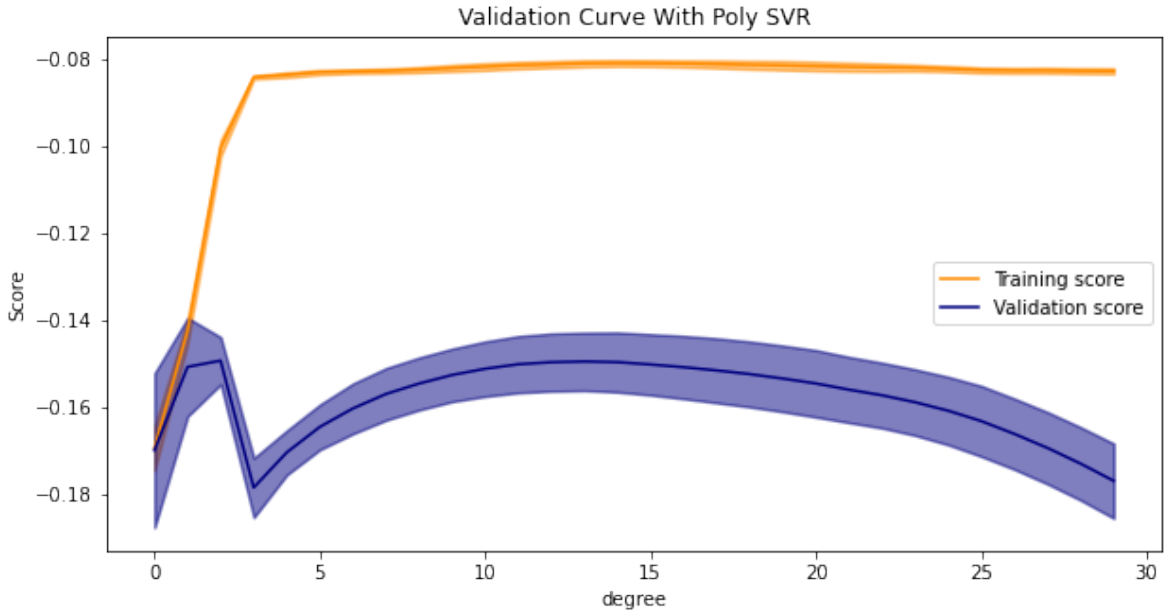
4.6 SVM z jądrem wielomianowym

W przypadku tego modelu - potrzebujemy wyznaczyć aż cztery hiperparametry:

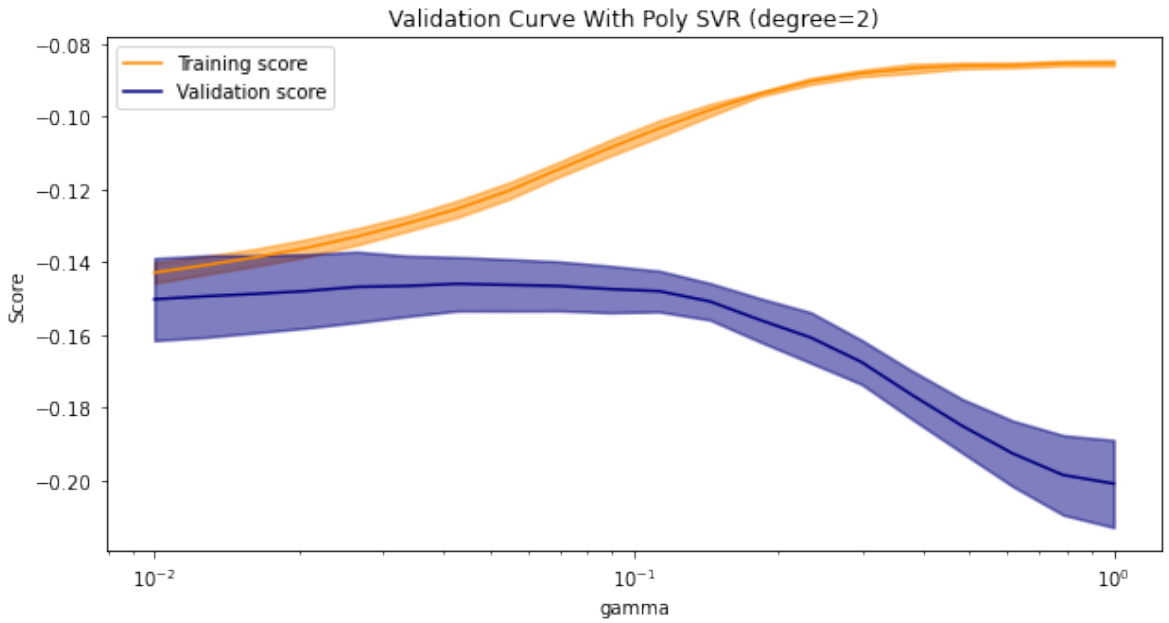
- Współczynnik wielomianu (degree, w zakresie od 0 do 30)
- Współczynnik dla jądra (gamma, w zakresie od 10^{-2} do 10^{0})
- Parametr regularyzacji (C, w zakresie od 10^{-3} do 10^{1})
- Współczynnik epsilon modelu (epsilon, w zakresie od 10^{-5} do 10^{1})

Kolejno testowaliśmy wartości dla następnych hiperparametrów, używając wcześniej wyznaczonych wartości jeśli już były dostępne, czyli np. przy liczeniu stopnia wielomianu inne wartości były arbitralnie wybrane, ale już przy liczeniu współczynnika gamma - stopień wielomianu został ustalony na ten wyznaczony w pierwszym szukaniu. Końcowo został stworzony model z wielomianem stopnia 2, wartością gamma ok. 0.043, wartością C ok. 0.89 i wartością epsilon ok. 0.002.

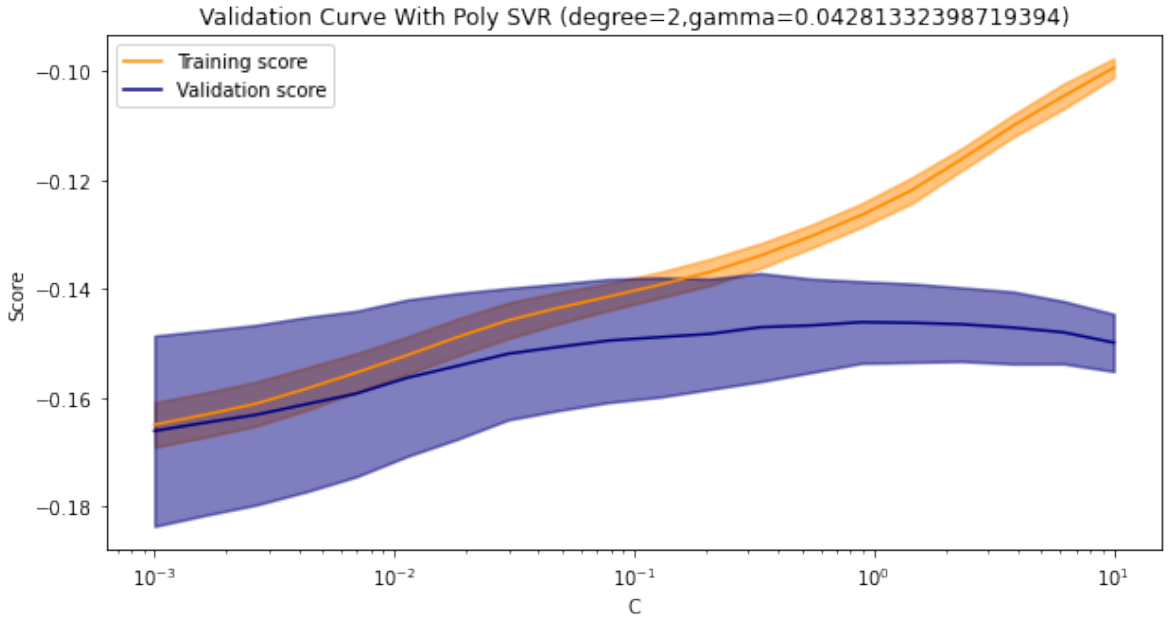
Otrzymane wyniki po raz pierwszy są faktycznie lepsze - błąd uczenia jest na poziomie ok. 8%, a błąd testowania na poziomie ok. 10.36%. Z wykresu krzywych uczenia możemy zauważyć, że model zyskałby na wprowadzeniu większej ilości danych. Głównym mankamentem jest jednak skalowalność modelu - typowo dla SVMów czas uczenia rośnie wraz z większą ilością danych.



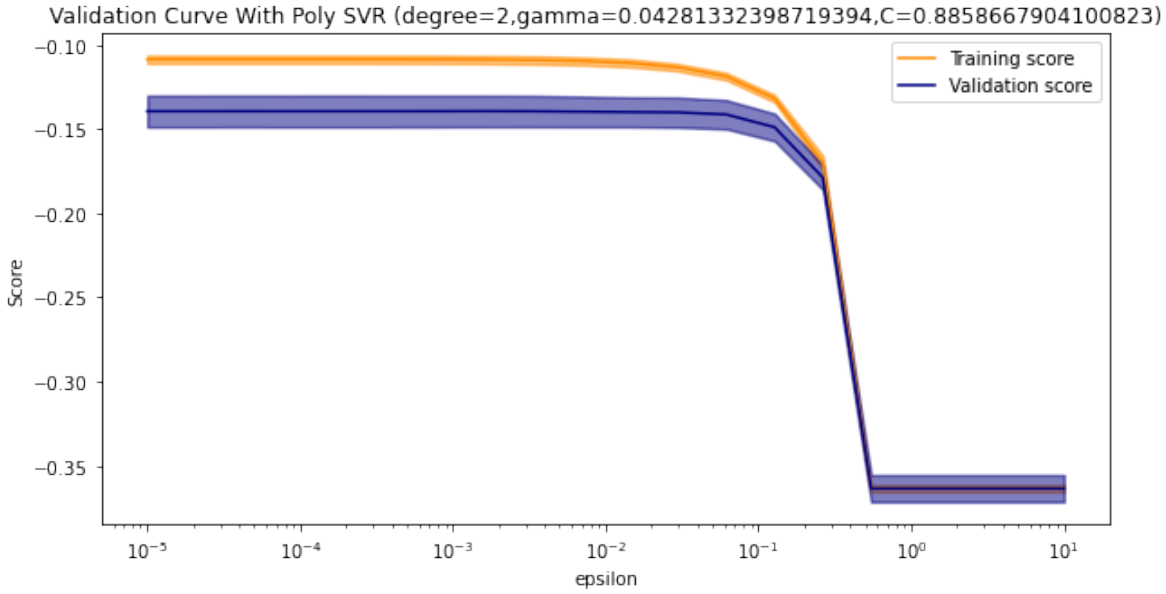
Rysunek 10: Wyniki w zależności od stopnia wielomianu



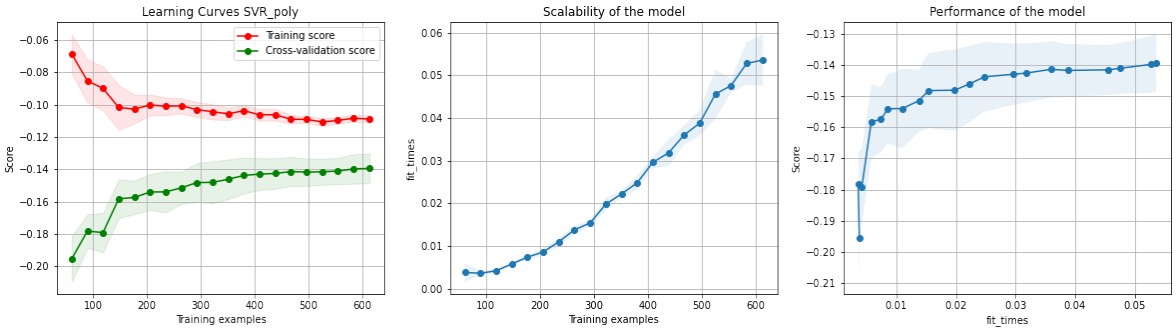
Rysunek 11: Wyniki w zależności od gamma



Rysunek 12: Wyniki w zależności od C



Rysunek 13: Wyniki w zależności od epsilon

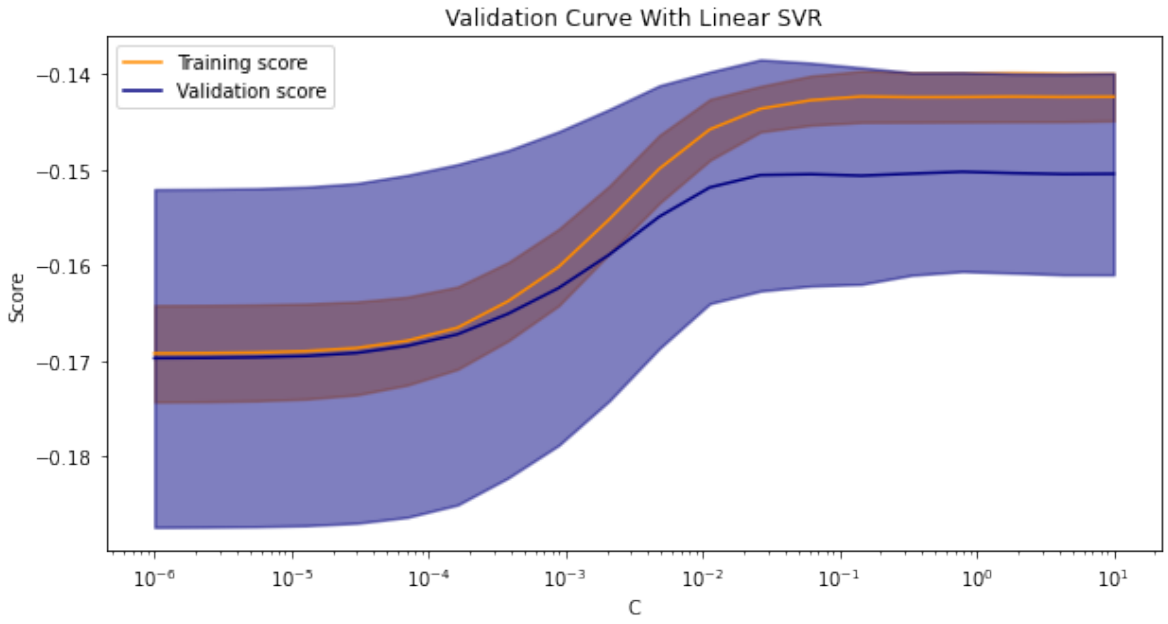


Rysunek 14: Krzywe uczenia, skalowalność modelu

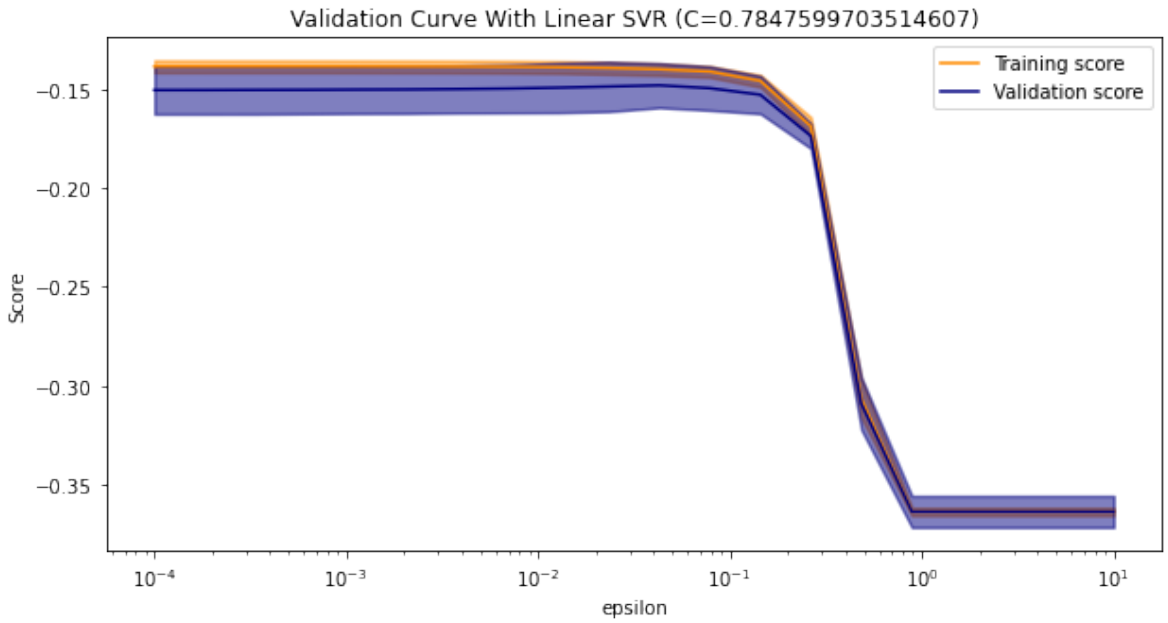
4.7 SVM z jądrem liniowym

W przypadku tego jądra musieliśmy wyznaczyć tylko wartość parametru regularyzacji C i wartość parametru epsilon. Wartość C szukana była w zakresie od 10^{-6} do 10^{1}, a wartość epsilon w zakresie od 10^{-4} do 10^{1}. Najlepsza wartość C wyniosła ok. 0.78, a wartość epsilon ok. 0.04.

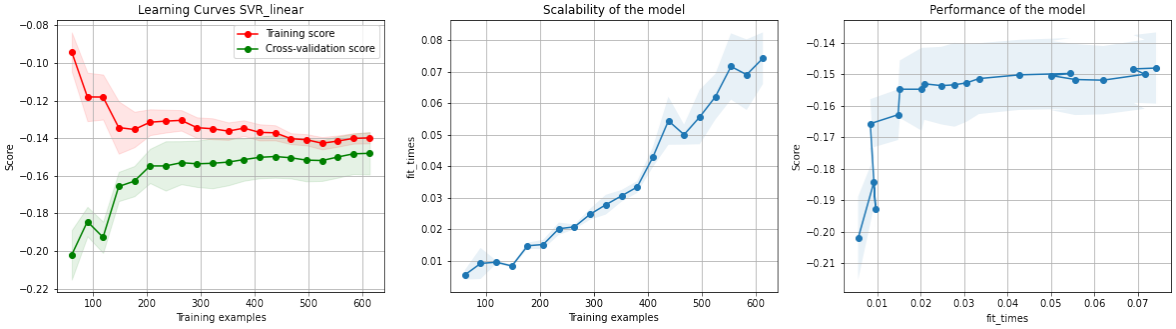
Model osiągnął wyniki gorsze niż przy jądrze wielomianowym - błąd uczenia wyniósł ok. 10.76%, a błąd testowania ok. 10.75%. Krzywe uczenia pokazują także, że model nie zyska zbytnio na zwiększeniu zbioru danych, a pozostałe wykresy pokazują do tego, że jest on dość słabo skalowalny.



Rysunek 15: Wyniki w zależności od C



Rysunek 16: Wyniki w zależności od epsilon

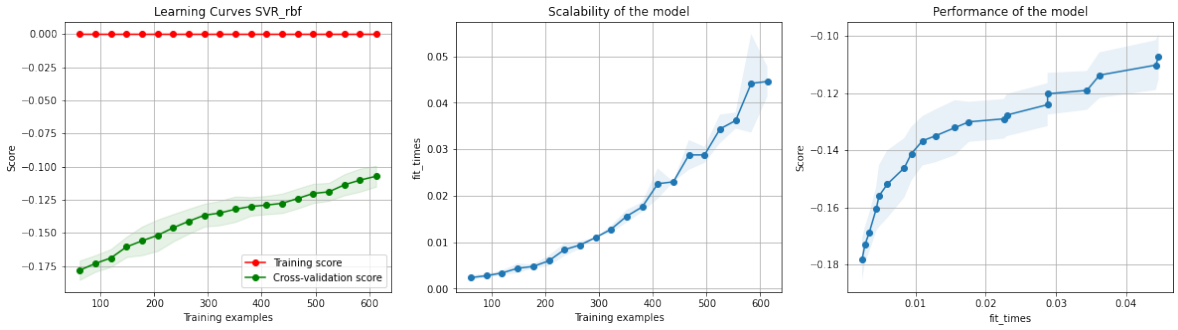


Rysunek 17: Krzywe uczenia, skalowalność modelu

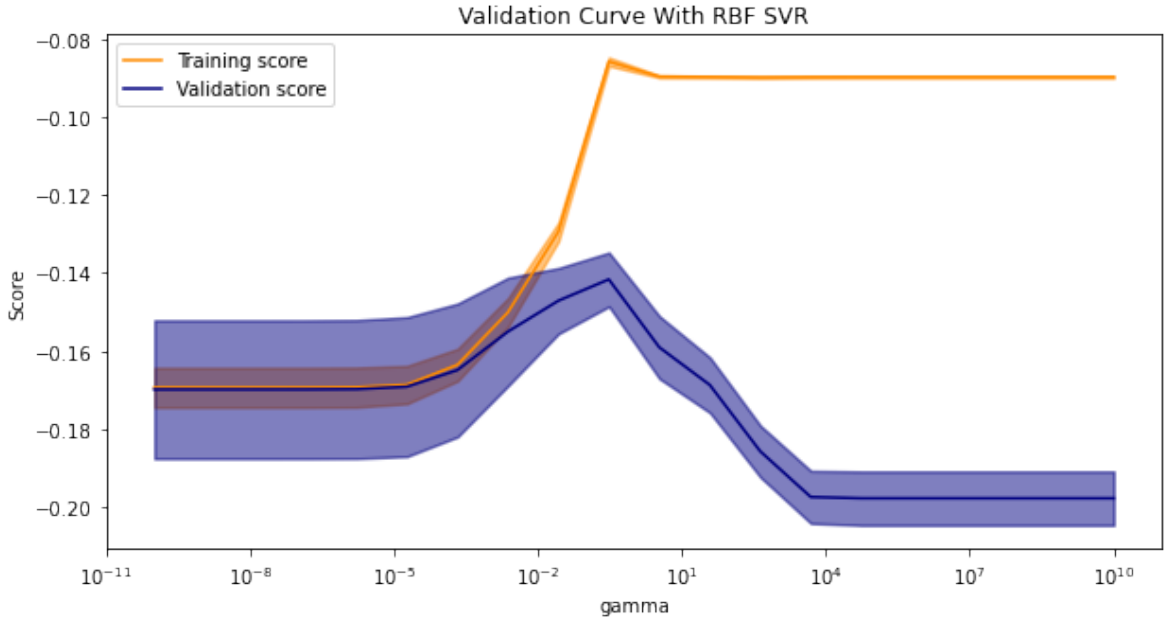
4.8 SVM z jądrem RBF

W przypadku tego jądra - potrzebowaliśmy te same parametry co w przypadku jądra wielomianowego, z wyjątkiem stopnia (a więc gamma, C i epsilon). Wartość gamma szukana była w zakresie od 10^{-11} do 10^10, Wartość C w zakresie od 10^{-4} do 10^3, a Wartość epsilon w zakresie od 10^{-5} do 10^2. Najlepsza wartość gamma wyniosła ok. 0.3, najlepsza wartość C okazała się górną granicą zakresu (1000), a najlepsza wartość epsilon - dolną granicą zakresu (0.00001).

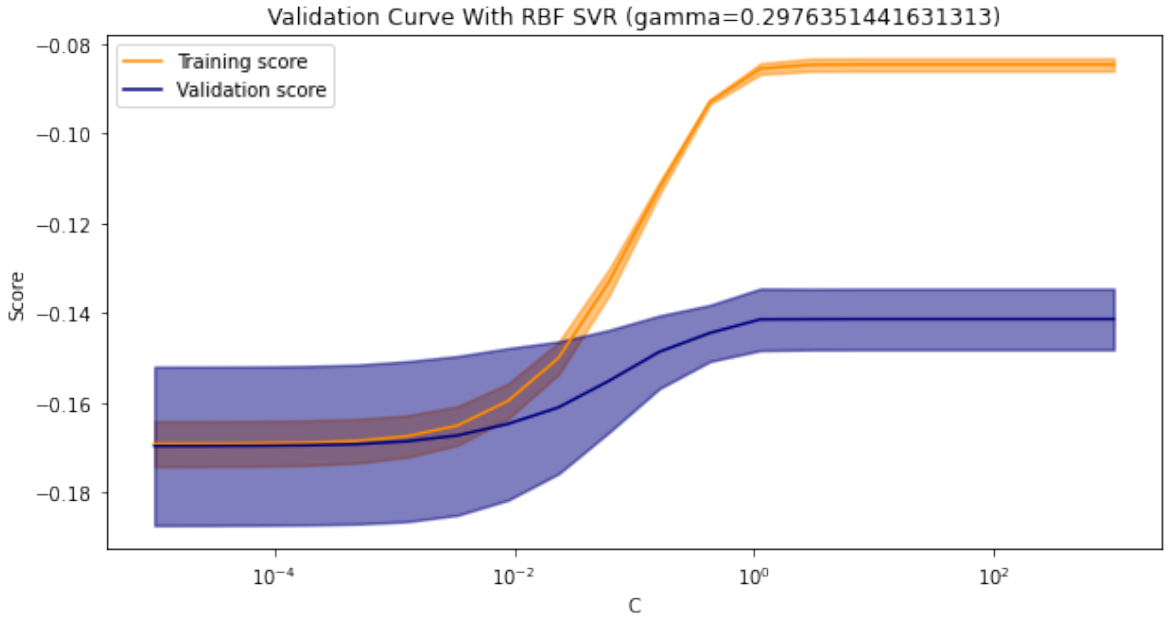
Model z jądrem RBF zdecydowanie przebił najlepsze dotychczasowe wyniki - błąd uczenia na poziomie ok. 0.01% (czyli właściwie żaden), a błąd testowania na poziomie 7.94%. Dodatkowo wykres krzywych uczenia pokazuje, że moglibyśmy poprawiać ten wynik zwiększając ilość danych. Jedynym problemem jest tu skalowalność - SVM z jądrem RBF jest dość intensywny obliczeniowo, co widać coraz bardziej w miarę dodawania danych.



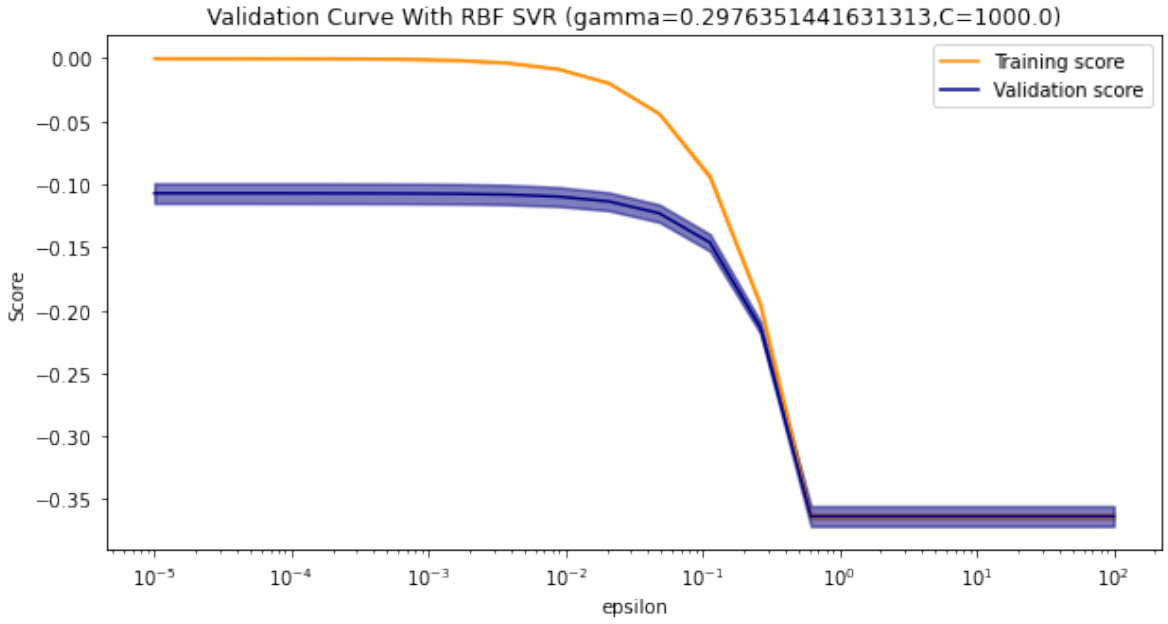
Rysunek 18: Krzywe uczenia, skalowalność modelu



Rysunek 19: Wyniki w zależności od gamma



Rysunek 20: Wyniki w zależności od C



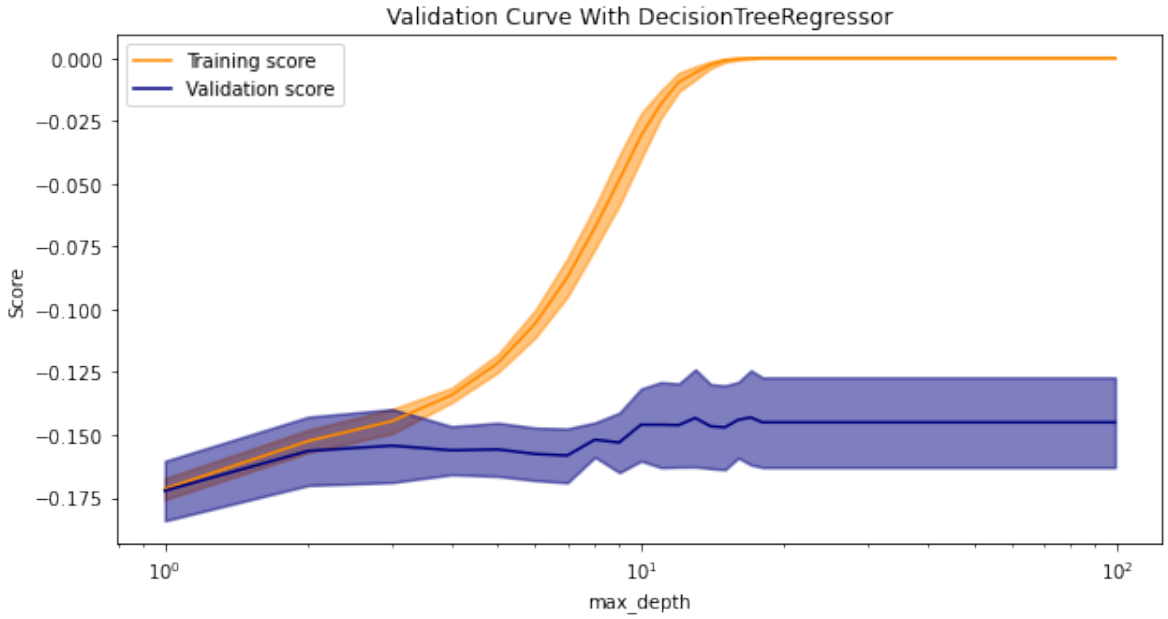
Rysunek 21: Wyniki w zależności od epsilon

4.9 Drzewo decyzyjne

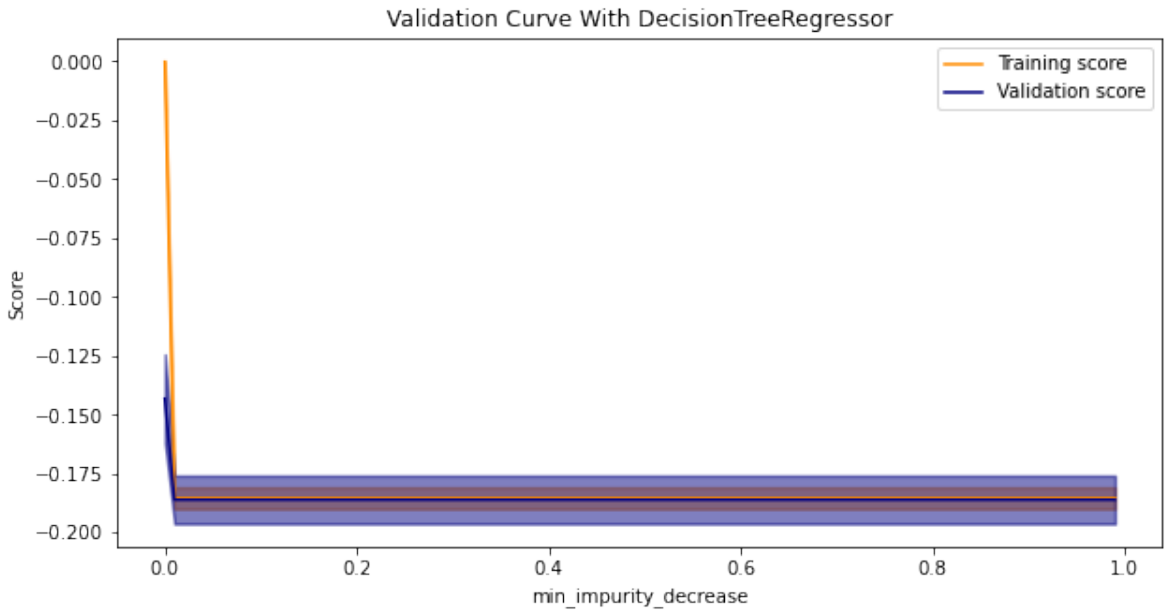
W przypadku tego modelu - potrzebujemy znowu wyznaczyć aż cztery hiperparametry:

- Maksymalna głębokość drzewa (max_depth, w zakresie od 0 do 100)
- Minimalny spadek zanieczyszczenia (min_impurity_decrease, w zakresie od 0 do 1)
- Maksymalna liczba liści (max_leaf_nodes, w zakresie od 0 do 100)
- Maksymalna liczba cech (max_features, w zakresie od 0 do 40)

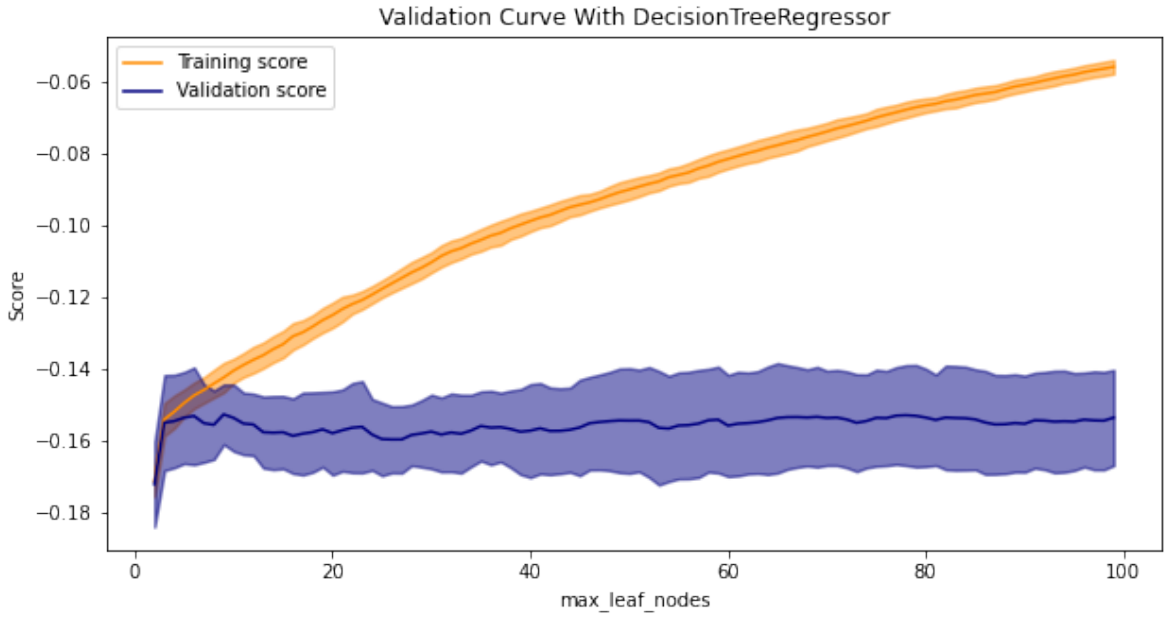
Testowanie parametrów przebiegało tak samo jak w przypadku wielomianowej SVM. Końcowo najlepsza głębokość drzewa wyniosła 17, spadek zanieczyszczenia 0, maksymalna liczba liści 9, a maksymalna liczba cech 32. Dla tych parametrów - błąd uczenia wyniósł ok. 11.6%, a błąd testowania ok. 13.9%. Model nie zyska zbytnio na dodaniu nowych danych, ale przynajmniej jest dość skalowalny. Jednak biorąc pod uwagę wszystkie wyniki - nie jest to dobry model w naszym przypadku.



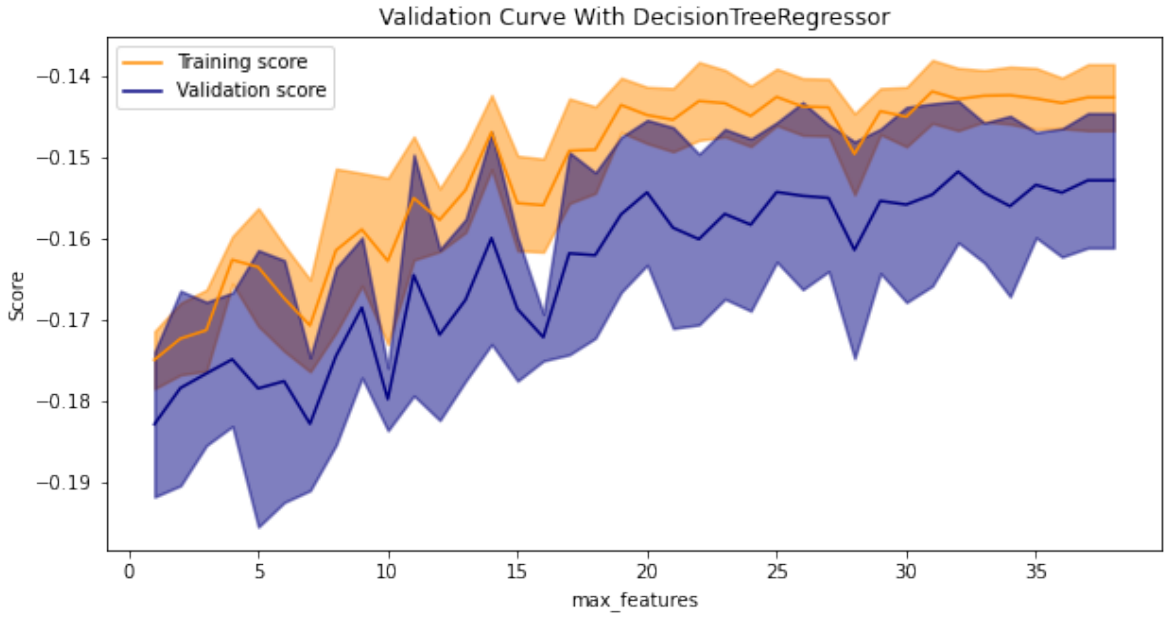
Rysunek 22: Wyniki w zależności od maksymalnej głębokości drzewa



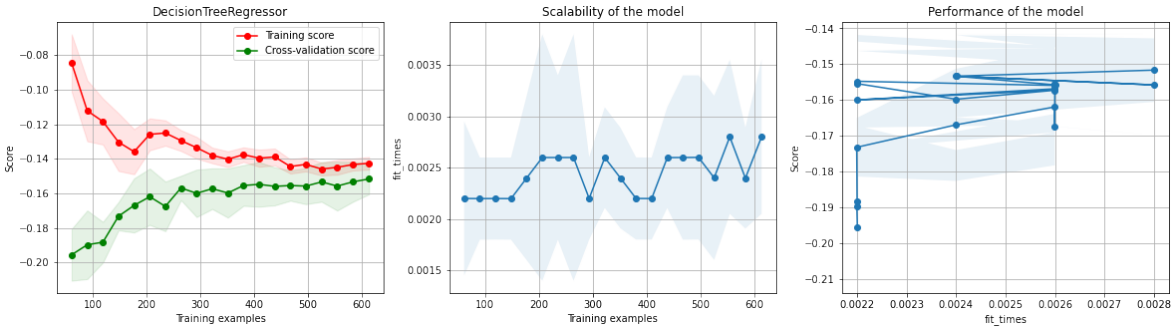
Rysunek 23: Wyniki w zależności od minimalnego spadku zanieczyszczenia



Rysunek 24: Wyniki w zależności od maksymalnej liczby liści



Rysunek 25: Wyniki w zależności od maksymalnej liczby cech



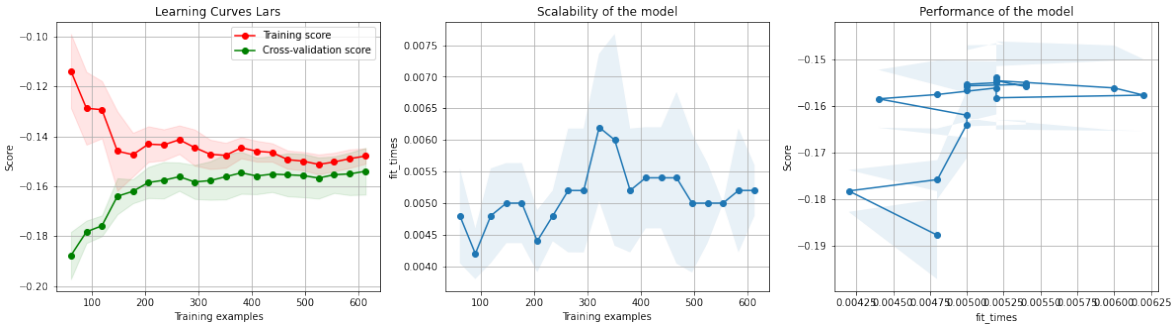
Rysunek 26: Krzywe uczenia, skalowalność modelu

4.10 Least angle regression

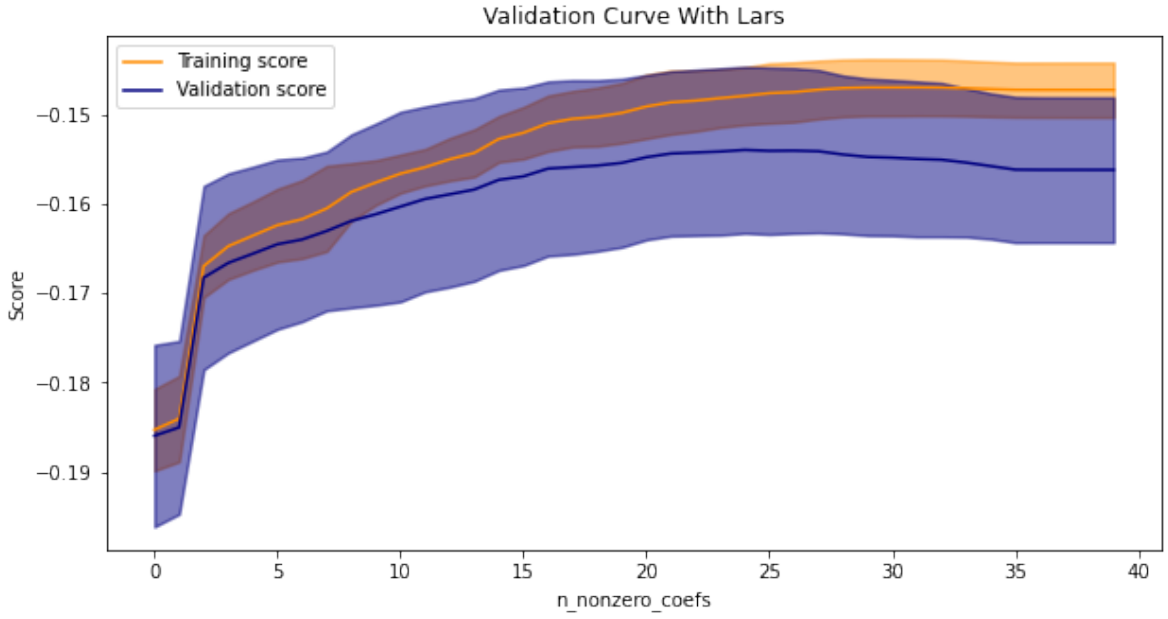
W przypadku tego modelu - potrzebujemy wyznaczyć trzy hiperparametry:

- Liczba niezerowych współczynników (n_nonzero_coefs, w zakresie od 0 do 40)
- Regularyzacja precyzji obliczeń (eps, w zakresie od 10^{-1} do 10^{2})
- Szum (jitter, w zakresie od 10^{-3} do 10^{-1})

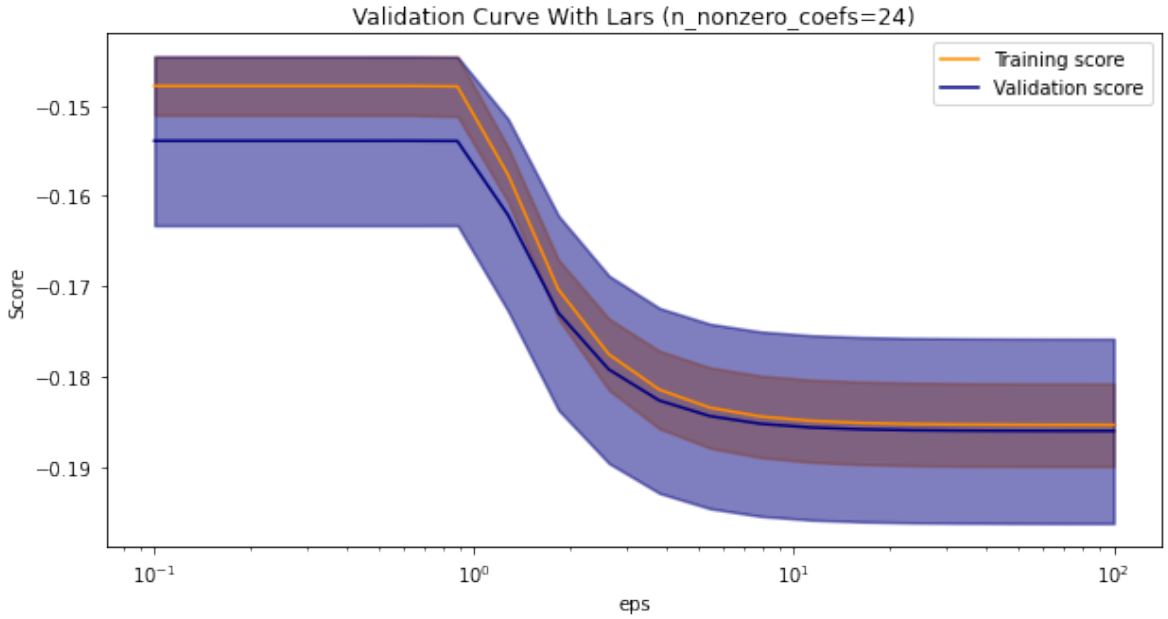
Najlepszą liczbą niezerowych współczynników okazała się wartość 24, najlepszą precyzją ok. 0.62, a najlepszym szumem wartość najmniejsza (0.001). Dla wyznaczonych parametrów błąd uczenia wyniósł ok. 11.89%, a błąd testowania ok. 11.75%. Model jest dobrze dość dobrze skalowalny i nie zyska zbytnio na zwiększeniu ilości danych. Biorąc to wszystko pod uwagę - model daje wyniki podobne do naszego dummy modelu.



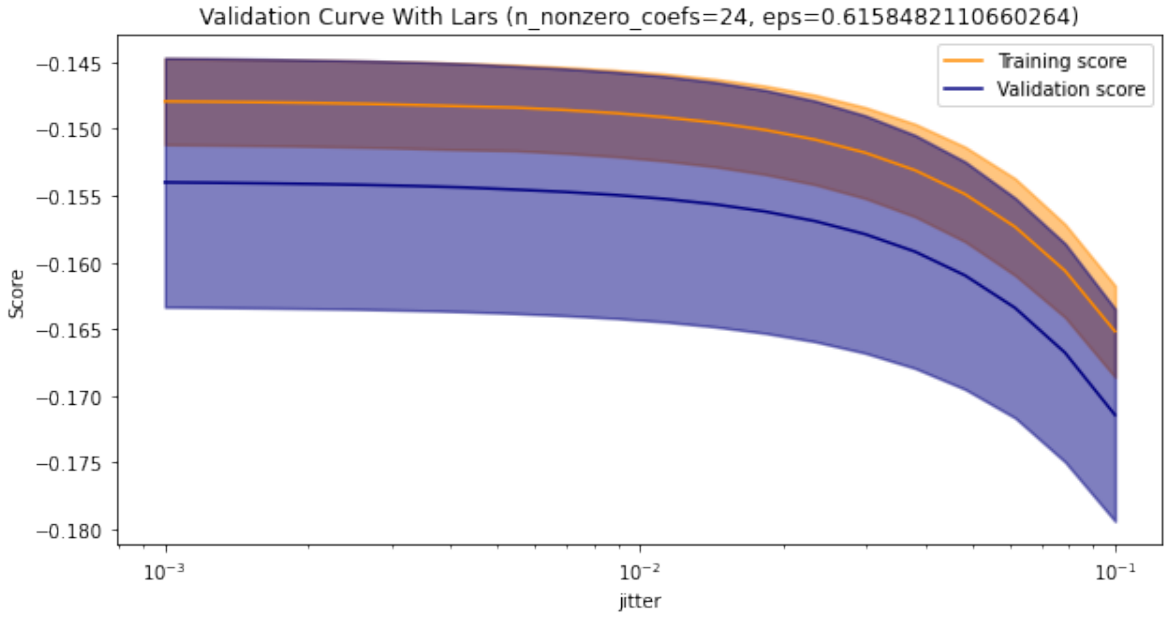
Rysunek 27: Krzywe uczenia, skalowalność modelu



Rysunek 28: Wyniki w zależności od liczby niezerowych współczynników



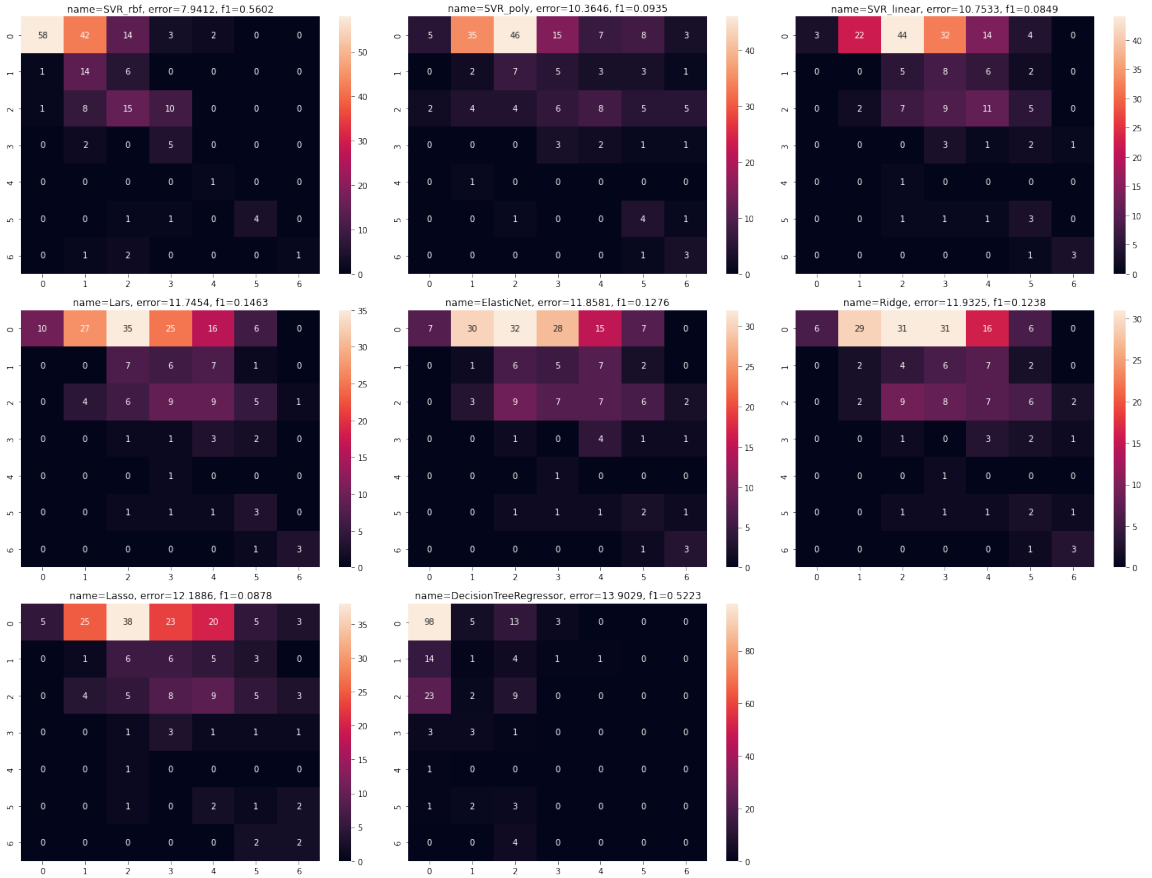
Rysunek 29: Wyniki w zależności od regularyzacji precyzji obliczeń



Rysunek 30: Wyniki w zależności od szumu

5 Macierz pomyłek

Po wszystkich testach - porównaliśmy nasze modele za pomocą macierzy pomyłek. Dla wyników została także wyznaczona wartość metryki F1. Modele regresji wielorakiej zostały pominięte.



Rysunek 31: Macierze pomyłek dla badanych modeli

6 Najlepszy Model

Najlepiej poradził sobie model SVM z jądrem RBF, z wynikiem MAPE równym niecałe 8% oraz "najzdrowszą" macierzą pomyłek.

7 Wnioski

Niektóre modele nie miały dostatecznej pojemności by zamodelować dane, a inne miały pojemność od nich znacznie większą i zyskałyby na zwiększeniu zbioru. Proste modele liniowe nie poradziły sobie z zadaniem. Najlepiej sprawdziły się modele z rodziny SVM z różnymi jądrami.

Błąd testowania najlepszego modelu (ok. 0.08) był na poziomie połowy różnicy między kolejnymi stopniami spożycia alkoholu - w danych oryginalnych spożycie alkoholu oceniane jest w pięciostopniowej skali, co po przeskalowaniu daje 0.2 na każdy stopień spożycia. Oznacza to, że nasz model nie powinien się mylić bardziej niż o pół stopnia w tej skali.

Największe błędy model osiąga dla osób, których warunki środowiskowe są bardzo trudne, ale osoby te mają silny charakter i nie przesadzają z alkoholem, pomimo trudnego życia (jest to bardzo optymistyczne spostrzeżenie). Poza tym model trochę myli także osoby niepijące z bardzo mało pijącymi (0 i 1), bardzo mało pijące z mało pijącymi (1 i 2) i osoby mało pijące ze średnio pijącymi (2 i 3). Jest to jednak problem regresji - nie typowej klasyfikacji.