Przewidywanie pozycji na mecie wyścigu kierowcy Formuły 1 na podstawie danych wyścigów z lat 2017-2023

Łukasz Wasilewski

12.06.2023

1 Wstęp

Mistrzostwa Formuły 1 to rywalizacja najszybszych kierowców na świecie jeżdżących w bolidach o niewyobrażalnych osiągach. Od roku 2017 w rywalizacji bierze udział 20 kierowców jeżdżacych dla 10 różnych zespołów. Sezon mistrzostw trwa rok i składa się z około 20 rund. Dzień przed każdym wyścigiem odbywa się sesja czasowa, która decyduje o ułożeniu stawki do startu wyścigu. Formuła 1 od lat cieszy się ogromnym zainteresowaniem kibiców na całym świecie. Co wyścig, widzowie z zaciekawieniem śledzą poczynania swoich ulubionych kierowców, którzy mogą liczyć na ich stały doping. Nic tak jednak nie nakręca spirali zainteresowania tym sportem jak bukmacherka oraz jakiekolwiek próby przewidywania wyników wyścigów. Nie ma osoby, która nie lubi uczucia posiadania racji, związanego z poprawnym wytypowaniem wyniku. W tym miejscu nasuwa się pytanie, czy na podstawie informacji o wyścigach z przeszłości realnie jesteśmy w stanie przewidywać na jakiej pozycji przyjedzie kierowca w danym wyścigu?

W mojej pracy zostanie podjęta próba sprawdzenia czy korzystając z danych wszystkich wyścigów odbytych w latach 2017-2023 jesteśmy w stanie wyliczyć pozycję na mecie kierowcy Formuły 1. Do badania będą wykorzystywane jedynie dane, które posiadamy przed momentem rozpoczęcia wyścigu takie jak pozycja kierowcy na starcie czy pozycja kierowcy w poprzednim wyścigu. Wyliczenia będą odbywać się za pomocą znanych modeli klasyfikacji wieloklasowej.

2 Zbiór danych i jego przetwarzanie

2.1 Zbiór danych

W pracy został wykorzystany jeden zbiór danych. Zbiór "Race results" pozyskany przy pomocy Ergast Developer API[1] zawiera dane o sezonie mistrzostw, rundzie mistrzostw, torze, kierowcy, wieku kierowcy, zespole kierowcy, pozycji startowej oraz pozycji na mecie.

2.2 Przetwarzanie wstępne

Zbiór danych nie posiadał pustych komórek, jednak zauważyłem, że w kilkunastu-kilkudziesięciu miejscach pozycja na starcie wynosi 0. Jest to wartość niedopuszczalna, gdyż kierowców jest 20 a pozycje zaczynają się od pierwszej. Po dłużej analizie danych doszedłem do wniosku, że głównie dotyczy to sytuacji, gdy kierowca startował do wyścigu z alei serwisowej. Start z alei serisowej jest rodzajem kary nakładanej zazwyczaj kiedy zespół wprowadził znaczące zmiany w bolidzie po zakończeniu sesji czasowej (wówczas jest to zabronione). Sytuacja ta jest równoznaczna ze startowaniem z końca stawki, dlatego postanowiłem wszystkie wartości 0 cechy "pozycja startowa" zamienić na wartość 20. Następnie za pomocą własnej metody do zbioru dodałem cechę "Pozycja w ostatnim wyścigu" mówiącą na jakiej pozycji kierowca skończył w ostatnim swoim wyścigu. Jeżeli dany kierowca debituje w wyścigu lub bierze udział pierwszy raz od roku 2017, w miejscu pozycji w ostatnim wyścigu jest wpisywana

wartość pozycji startowej. Dodatkowo pobrałem plik zawierający wyniki z ostaniego wyścigu w 2016 roku, aby uniknąć uzupałnienia samymi sztucznymi danymi pierwszego wyścigu roku 2017. Na końcu usunąłem dane, które nie wnosiły nic do dalszej analizy problemu czyli sezon, rundę, tor i kierowcę.

2.3 Analiza eksploracyjna

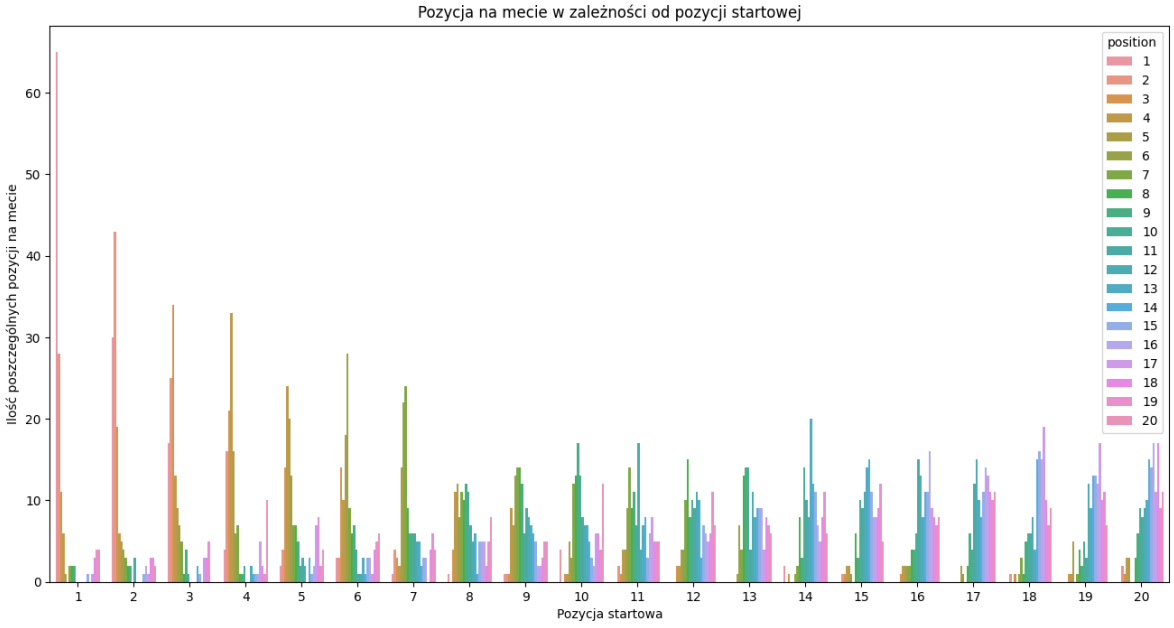
Zbiór zawiera dane 2600 wyników wyścigów. Poniższa tabela prezentuje dane po wstępnym przetworzeniu.

Wiek	Zespół	Pozycja w ostatnim wyścigu	Pozycja startowa	Pozycja końcowa
29	ferrari	3	2	1
32	mercedes	1	1	2
27	mercedes	20	3	3
37	ferrari	6	4	4
19	red_bull	4	5	5
27	williams	14	18	16
23	mclaren	9	3	17
30	haas	19	17	18
33	alfa	11	16	19
22	williams	18	20	20

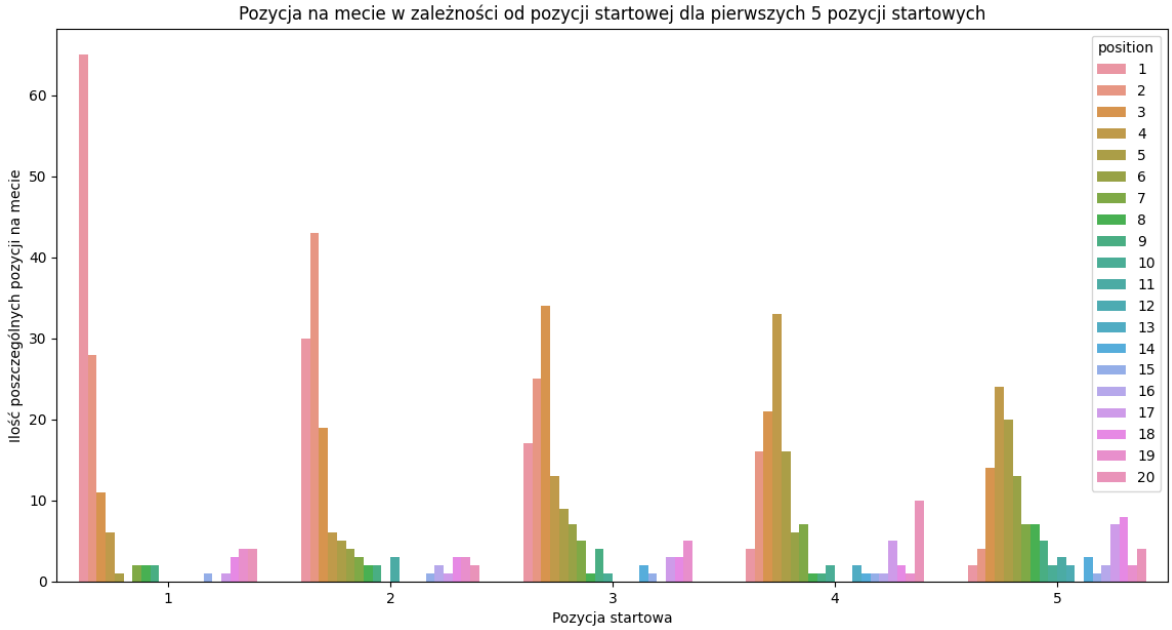
Tabela 1: Tabela prezentująca wycinek danych po wstępnym przetworzeniu

Analiza poszczególnych parametrów:

Analiza pozycji startowej



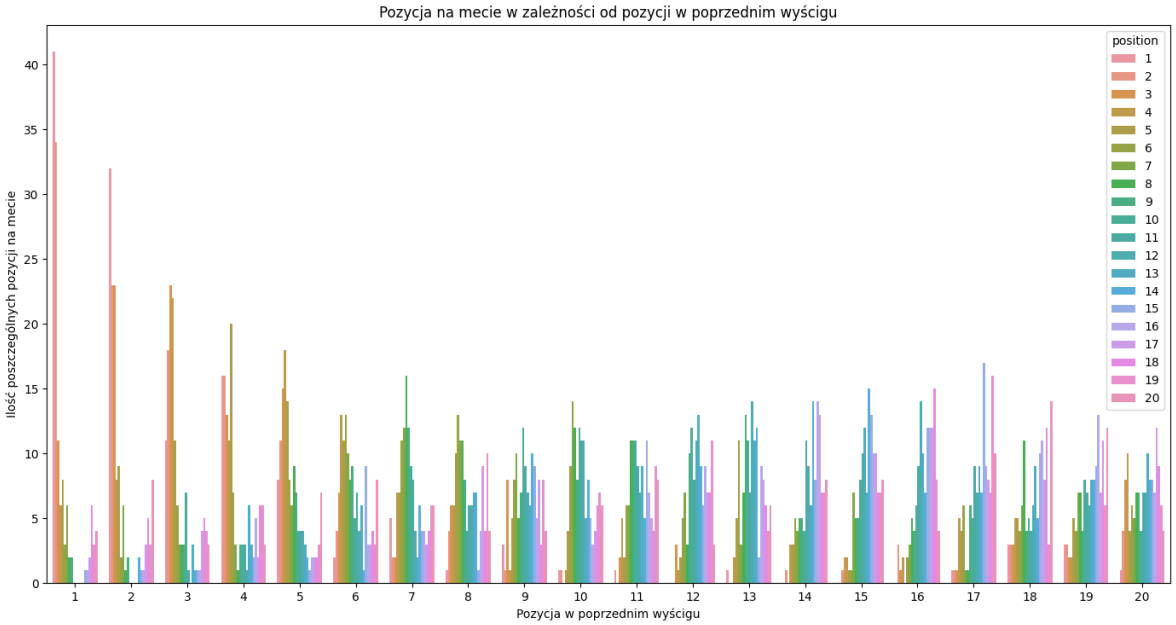
Rysunek 1: Pozycja na mecie w zależności od pozycji startowej



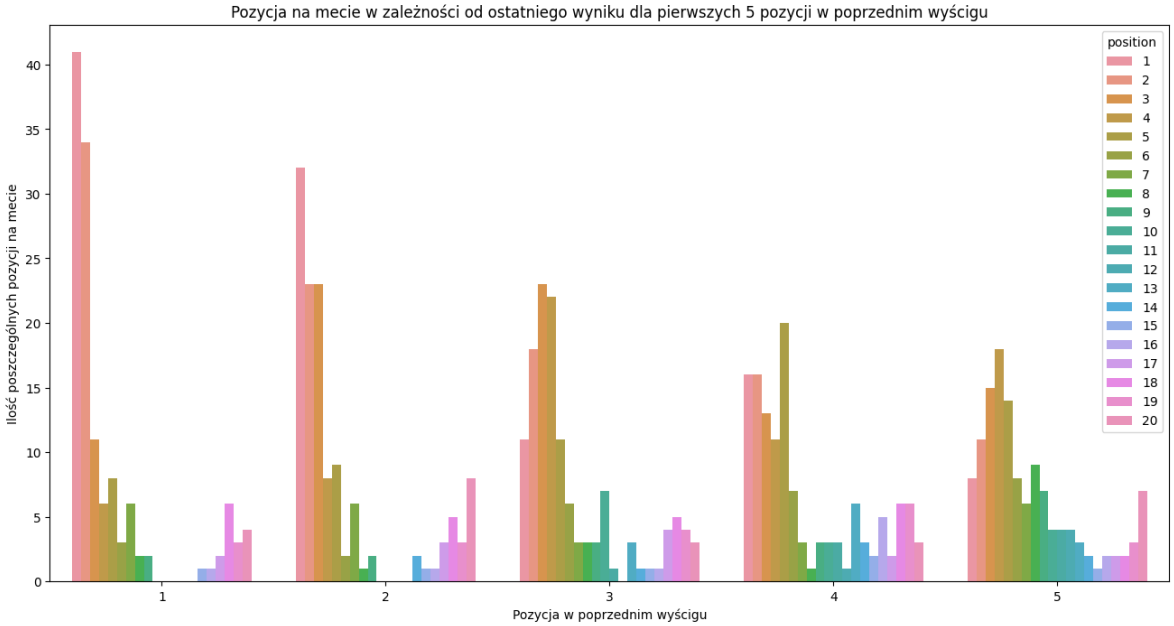
Rysunek 2: Pozycja na mecie w zależności od pozycji startowej dla pierwszych 5 pozycji startowych

Jak można zauważyć na wykresach, wyższa pozycja startowa znacząco przyczynia się do wyższej pozycji na mecie.

Analiza pozycji w ostatnim wyścigu



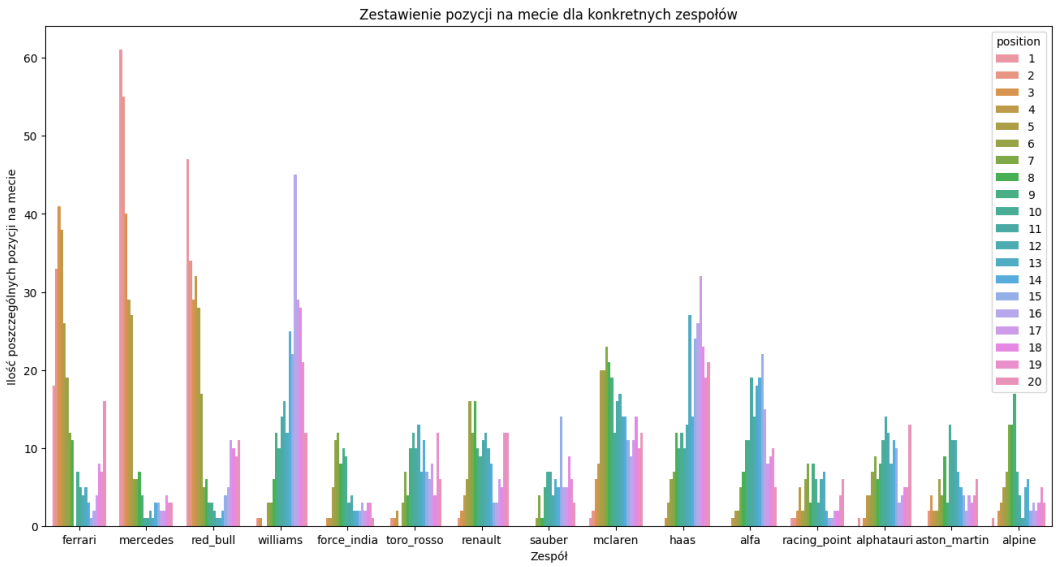
Rysunek 3: Pozycja na mecie w zależności od pozycji w poprzednim wyścigu



Rysunek 4: Pozycja na mecie w zależności od ostatniego wyniku dla pierwszych 5 pozycji w poprzednim wyścigu

Tak jak dla pozycji startowej, im wyższa pozycja w poprzednim wyścigu tym wyższa pozycja na mecie.

Analiza zespołu

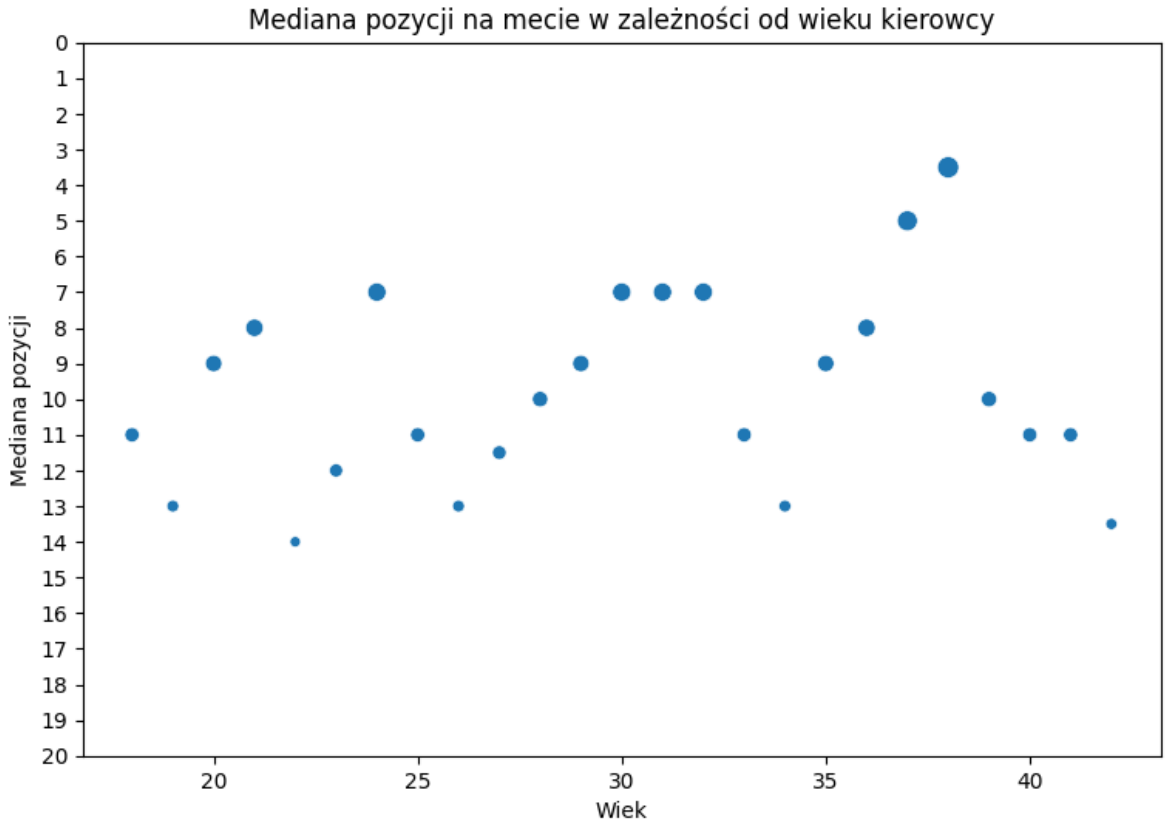


Rysunek 5: Zestawienie pozycji na mecie dla konkretnych zespołów

Zespół dla którego jeżdżą kierowcy również ma duże znaczenie dla wyniku końcowego. Zespoły

czołowe takie jak Ferrari, Mercedes i Red Bull zazwyczaj zajmują najwyższe lokaty. Ewentualne ich słabsze wyniki są spowodowane sytuacjami losowymi. Williams i Haas za to są najsłabszymi zespołami. Ich lepsze wyniki mogą być jedynie spowodowane słabszą dyspozycją czołówki. Niektóre zespoły mają łącznie mniej danych, gdyż pomiędzy sezonami mogą nastąpić zmiany w konstrukcji stawki, np. inny koncern przejmuje czyiś zespół lub jeden zespół upada a nowy dołącza do mistrzostw. Mimo wszystko od sezonu 2017 łączna liczba zespołów w stawce pozostaje niezmienna czyli 10.

Analiza wieku



Rysunek 6: Mediana pozycji na mecie w zależności od wieku kierowcy

Najwyższy środkowy wynik mają kierowcy w wieku 38 lat, najniższy zaś kierowcy w wieku 22 lat. Kierowcy w wieku średnim (30-32) jeżdżą na solidnym, stabilnym poziomie. Można byłoby się spodziawać bardziej regularnego wykresu, lecz już tłumaczę czemu jest jaki jest. Teoretycznie kierowcy najsłabsze wyniki powinni osiągać na początku oraz u schyłku swojej kariery. I faktycznie tak jest, z tym że są kierowcy, którzy odchodzą z Formuły 1 już wieku 25 lat i nigdy do niej oraz są też kierowcy którzy mają 40 lat i nie zapowiada się żeby prędko mieli odejść. Dodatkowo inne czynniki takie jak chociażby zespół mają po prostu większy wpływ na osiągane rezaltaty.

3 Eksperymenty

Do eksperymentów wykorzystałem modele klasyfikacji z biblioteki scikit-learn.

Dla każdego eksperymentu przewiduję następujące statystyki reprezentujące jakość przprowadzonych obliczeń:

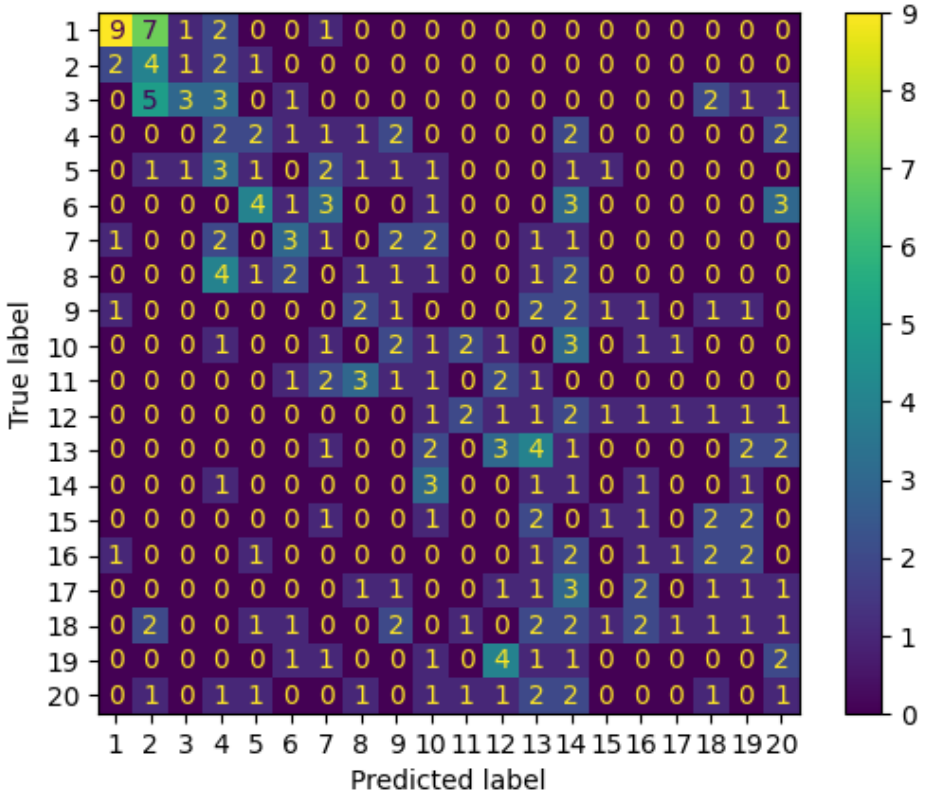
- Macierz pomyłek - graficzne zestawienie prawdziwych wartości z wartościami wyliczonymi przez model,
- Dokładność - statystyka będąca ilorazem liczby poprawnie sklasyfikowanych próbek i liczby wszystkich próbek,
- Zmodyfikowana dokładność - autorska wariacja dokładności, w której do liczby poprawnie sklasyfikowanych próbek dodajemy liczbę prawie poprawnie sklasyfikowanych próbek (takich które różniły się o jedną klasę w górę lub w dół) pomnożoną razy 0,5.

3.1 Klasyfikacja z wykorzystaniem modelu SVM

Pierwszym użytym przeze mnie modelem do klasyfikacji jest SVM. Model ten dobrze sobie radzi w przypadku danych o wysokiej wymiarowości, oraz obsługuje wieloklasową klasyfikację.

Parametry modelu:

kernel	gamma	C
rbf	10	1



Rysunek 7: Macierz pomyłek dla klasyfikacji z użyciem SVM

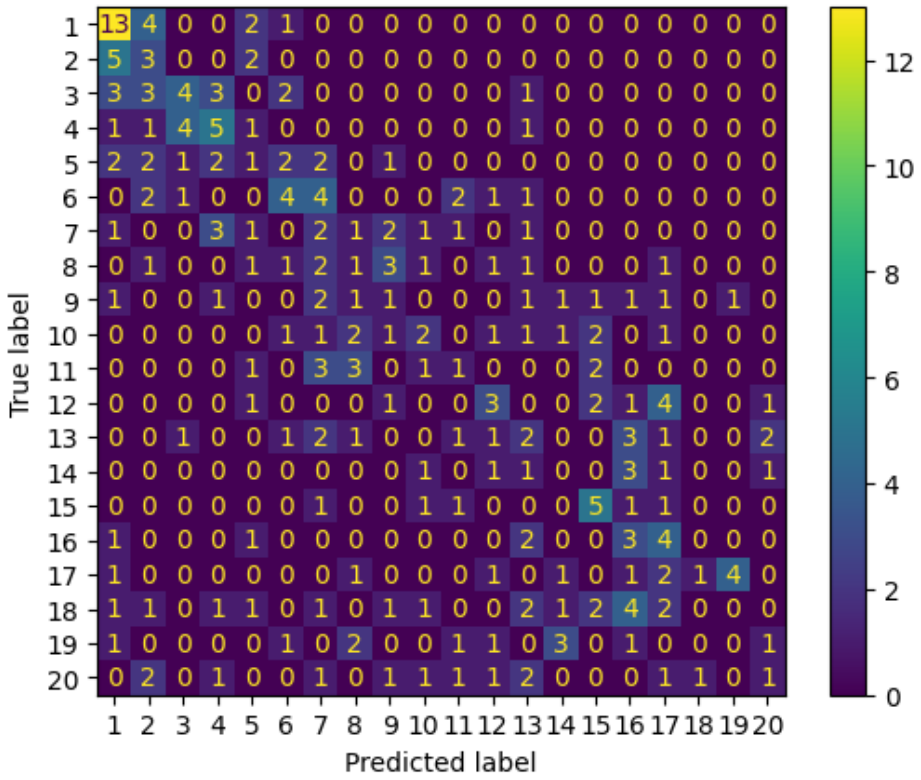
Dokładność: 0,131
Zmodyfikowana dokładność: 0,246

3.2 Klasyfikacja SVM z innymi parametrami

Ponownie wykonamy klasyfikację z wykorzystaniem SVM tym razem jednak z innymi parametrami. Zmniejszając wartość gamma powinniśmy móc rozluźnić nieco model. Zwiększając wartość C powinniśmy za to zminimalizować błąd dopasowania.

Parametry modelu:

kernel	gamma	C
rbf	0.05	10



Rysunek 8: Macierz pomyłek dla klasyfikacji z użyciem SVM z innymi parametrami

Dokładność: 0,204
Zmodyfikowana dokładność: 0,298

Wyniki znacząco się poprawiły.

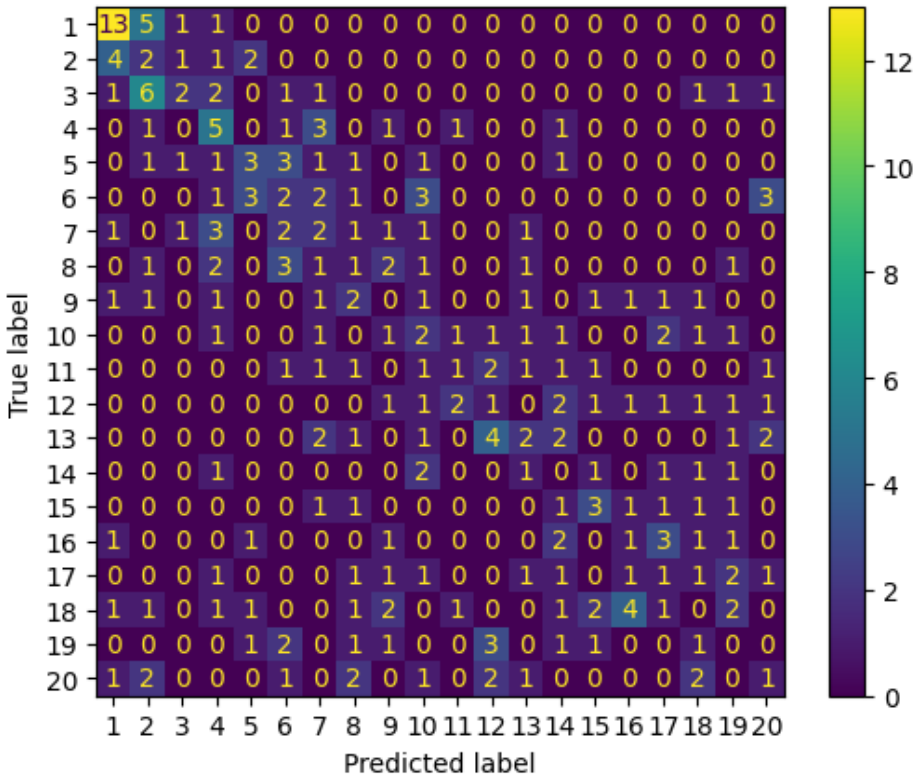
3.3 Klasyfikacja z wykorzystaniem modelu Random Forrest

Kolejnym modelem, który chciałem wykorzystać jest Random Forrest. Zdecydowałem się go wybrać, ponieważ jest on algorytmem opartym na złożeniu wielu drzew decyzyjnych. Dzięki temu może radzić

sobie z różnorodnymi zestawami danych, a także redukować efekt przeuczenia. W przypadku mojego problemu z 4 cechami i 20 klasami, Random Forest może budować wiele drzew decyzyjnych, które uwzględniają różne kombinacje cech i tworzą złożone granice decyzyjne.

Parametry modelu:

liczba drzew	maksymalna głębokość
100	None



Rysunek 9: Macierz pomyłek dla klasyfikacji z użyciem Random Forrest

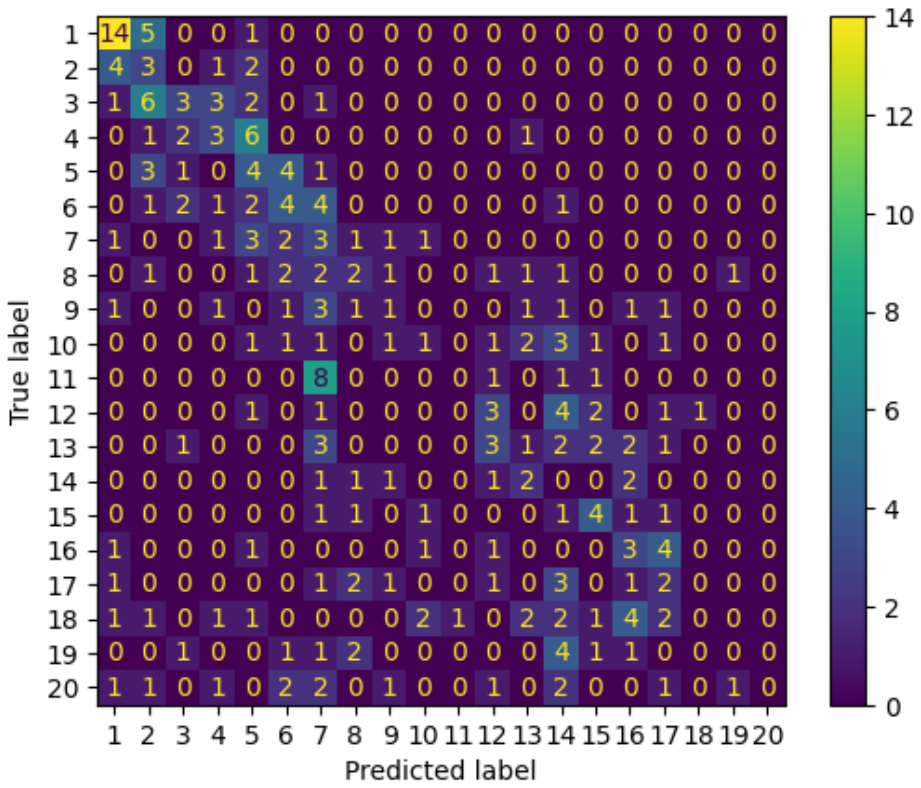
Dokładność: 0,162
Zmodyfikowana dokładność: 0,281

3.4 Klasyfikacja Random Forrest z innymi parametrami

Ponownie wykonamy klasyfikację z wykorzystaniem Random Forrest. Tym razem ustawimy maksymalną głębokość drzewa oraz lekko zwiększymy ilość drzew.

Parametry modelu:

liczba drzew	maksymalna głębokość
150	5



Rysunek 10: Macierz pomyłek dla klasyfikacji z użyciem Random Forrest z innymi parametrami

Dokładność: 0,195
Zmodyfikowana dokładność: 0,315

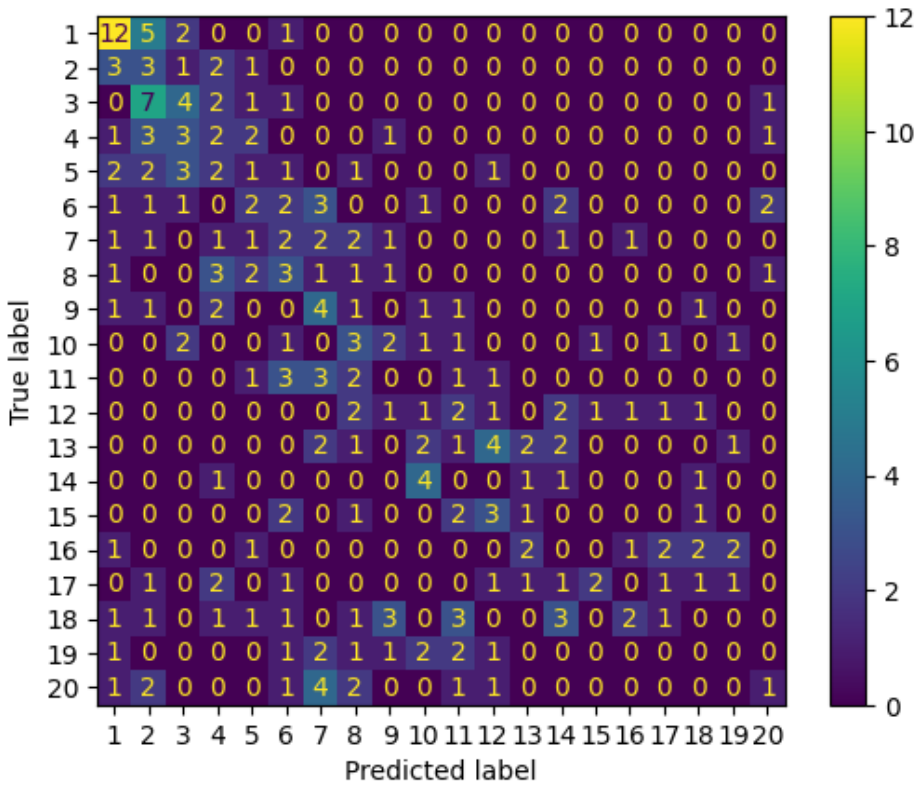
Wyniki lekko się poprawiły.

3.5 Klasyfikacja z wykorzystaniem modelu K-Nearest Neighbors

Ostatnim modelem, który postanowiłem wykorzystać jest K-Nearest Neighbors. Jest to prosty model szczególnie przydatny, gdy dane mają lokalne podobieństwo. W przypadku, gdy obiekty o podobnych cechach mają tendencję do należenia do tych samych klas, K-NN może być bardzo skutecznym modelem klasyfikacyjnym. Uważam, że jest to ciekawy wybór, który może dużo powiedzieć o stopniu zaawansowania mojego problemu.

Parametry modelu:

liczba sąsiadów	waga
5	uniform



Rysunek 11: Macierz pomyłek dla klasyfikacji z użyciem K-Nearest Neighbors

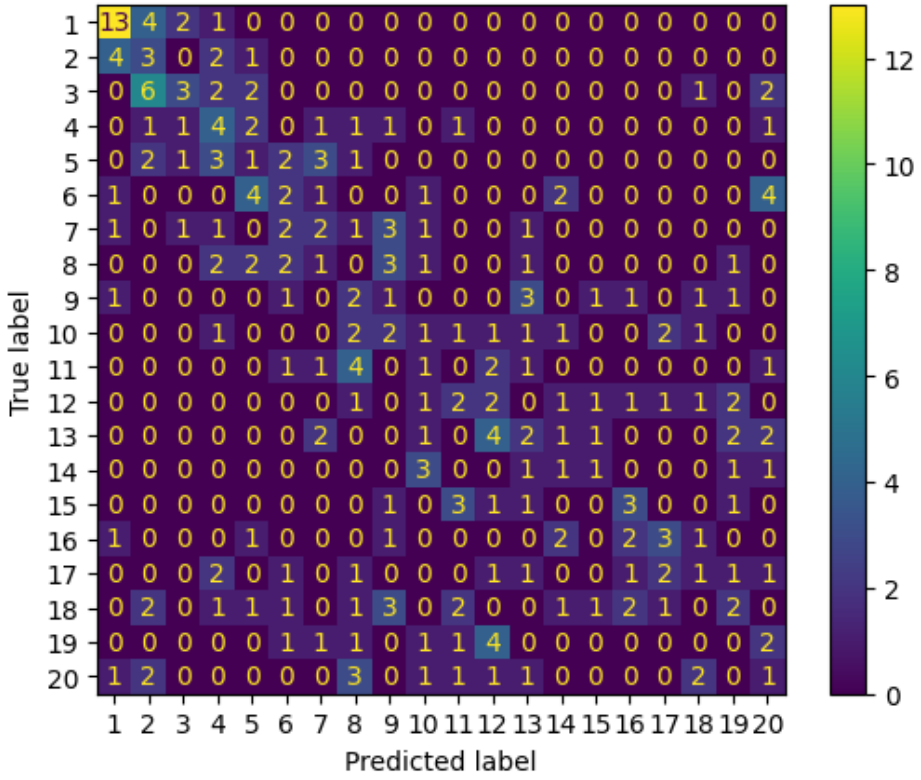
Dokładność: 0,138
Zmodyfikowana dokładność: 0,246

3.6 Klasyfikacja K-Nearest Neighbors z innymi parametrami

Ponownie wykonamy klasyfikację z wykorzystaniem K-Nearest Neighbors. Tym razem zwiększymy nieco liczbę sąsiadów oraz zmienimy wagę z takiej, w której wszystkie sąsiedztwa mają ten sam wpływ na taką, w której bliższe sąsiedztwa mają większy wpływ na klasyfikację.

Parametry modelu:

liczba sąsiadów	waga
11	distance



Rysunek 12: Macierz pomyłek dla klasyfikacji z użyciem K-Nearest Neighbors z innymi parametrami

Dokładność: 0,154
Zmodyfikowana dokładność: 0,281

Wyniki trochę się poprawiły

4 Wnioski

Model	Dokładność	Z.Dokładność
SVM 1	0,131	0,246
SVM 2	0,204	0,298
Random Forrest 1	0,162	0,281
Random Forrest 2	0,195	0,315
K-Nearest Neighbors 1	0,138	0,246
K-Nearest Neighbors 2	0,154	0,281

Tabela 2: Tabela prezentująca wyniki dokładności dla zastosowanych modeli

Przedstawione eksperymenty pokazały, że w pewnym stopniu jesteśmy w stanie wyliczyć, na której pozycji przyjedzie kierowca uwzględniając wyniki z poprzednich wyścigów. Na pewno sporym kłopetem przy klasyfikacji była duża ilość klas. Gdyby pogrupować pozycje w ogólniejsze klasy modele na pewno lepiej by sobie poradziły, natomiast odbyłoby się to kosztem precyzji wyników. Trzeba pamiętać, że w Formule 1 jak w każdym sporcie dużo zależy od przypadku. Pewnych sytuacji takich jak awaria bolidu

czy wypadek nie jest w stanie przewidzieć ani człowiek ani żadna maszyna. Co do samych modeli najlepiej z problemem poradziły sobie SVM oraz Random Forrest. SVM był najdokładniejszy, lecz Random Forrest nadrobił drobnymi pomyłkami. W przypadku SVM możemy zauważyć, że dobranie odpowiednich parametrów może bardzo mocno wpłynąć na jakość modelu. K-Nearest Neighbors natomiast okazał się najsłabszym wyborem dla tego problemu, prawdobodobnie z powodu różnych wyników dla podobnych cech. Jedną z ciekwszych obserwacji wydaje się to, że modele najmniejszy problem miały z przewidzeniem pierwszej pozycji. To pokazuje, że najprościej w Formule 1 jest przewidzieć kto wygra wyścig, a reszta pozycji pozostaje małą tajemnicą.

Literatura

[1] "Ergast Developer API", http://ergast.com/mrd/, dostęp: 12.06.2023.