

Plotly Python Dash

February 2022

1 Binär klassificering

Visualisering av hur era välbekanta klassificeringsalgoritmer faktiskt jobbar! Förhoppningsvis får ni lite svar på VAD vissa parametrar faktiskt bidrar med. Med hjälp av Plotly DASH kan man göra interaktiva plots med sliders och dropdownval som automatiskt renderar resultaten av parameterintervall.

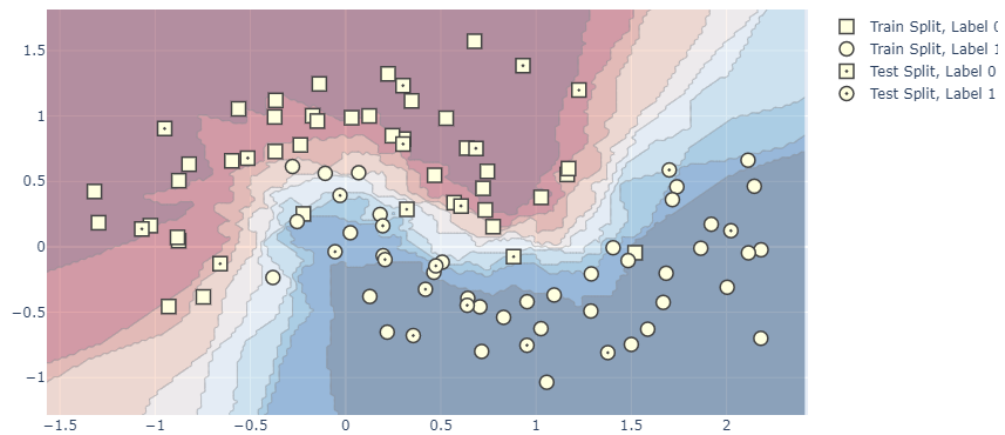


Figure 1: kNN beslutsregion för binär klassificering.

Core components i Dash hittar man [i den här länken](#) och behövs för att förstå syntaxen lite och hur man knyter samman input från interaktivitet med en machine learningmodell i backenden.

Använd valfritt exempel gjorda i matplotlib från [den här länken](#) för att skapa ett interaktivt lager som ger inargument till en modell på samma sätt som i [den här länken](#) att översätta från matplotlib till Plotly.

Använd en (eller flera) slider(s) på samma sätt som i exemplet, men ersätt kNN som classifier med så många modeller ni har lust med. Kan

Jämför contour plots mellan kNN, decision tree classifier och random forest. Hur skiljer sig färggradienterna åt och vad innebär det? Tänk på vad färgschemat säger och hur de olika algoritmerna fattar beslut för klassificering. Hur utvärderas varje punkt i meshen (det diskreta rutnätet i bilden) för respektive punkt?

Testa att i kNN ändra antalet närmsta grannar till 1. Vad säger det dig? Hur relaterar du detta till overfitting?

Gör states istället för input. Händer det något speciellt när du interagerar med state baren?

1.1 Lite svårare

Se till att inputen till din modell inte behöver komma i argumentordning, utan att den istället ska kunna skickas in som key word arguments.

Gör en SVM-modell som kan ta emot vilken kernel, vilket C och vilket gamma som helst.

Sätt polynomial kernel till 1. Hur skiljer det sig från linear kernel? Varför?

2 Regression

Behöver lite mer kärlek. Kan uteslutas i nuläget.