Daniel Endresen gruppe 41, 15.11.2024

SCOPE

Målet med prosjektet er å predikere vin kvaliteten til en gitt rødvin på en skala fra 1-10. Prediksjonen er mulig ved å bruke de forskjellige kjemiske og fysiske egenskapene til vinen, som eksempelvis kan være syrlighet, alkoholprosent, og sukkernivå. Modellen vil kunne hjelpe vinprodusenter å kunne evaluere kvaliteten til produktene sine, med en data drevet prosess. Dette vil føre til at det blir mindre ledd i produksjonsprosessen og bidra til bedre kvalitetssikring for rødvinsproduksjon.

Løsningen som har blitt brukt i dette prosjektet vil bli brukt som et verktøy for å predikere vin kvalitet basert på data som blir gitt av bruker. Bruker vil kunne benytte seg av en web basert plattform hvor de kan plotte inn de ulike målingene til vinen sin, og deretter få en estimert tilbakemelding av kvaliteten til produktet på en skala fra 1-10.

For øyeblikket er det fortsatt eksperter som blir brukt til å evaluere vin. Dette er meget nøyaktig, men det er subjektivt, tidskrevende og dyrt. En automatisert maskinlæringsprosess vil løse dette, men vil ikke være like nøyaktig.

For å teste ytelsen til maskinlæringsmodellen har det blitt tatt i bruk nøyaktighetsmåling for å se hvor nøyaktig hver modell kan gjette på de ulike klassene. Modell presisjon og recall har også blitt brukt for å sjekke hvordan modellen oppfører seg på tvers av klassene, eksempelvis dårlig og god kvalitet.

DATA

Dataen brukt i prosjektet er kjemiske egenskaper av rødvin. Disse egenskapene er representert som numeriske features : Syrlighet, sukkernivå, cholorider, svoveldioksid, tetthet, pH og alkohol. Målet er å predikere kvaliteten til vinen ved hjelp av disse kjemiske verdiene. Datasettet som har blitt brukt er tilgjengelig fra UCI Machine Learning Repository - Wine Quality Dataset.

Datasettet for rødvin inneholder 1599 rader med komplette vinmålinger. Dette er på den mindre siden, men kan fungere for enklere maskinlæringsmodeller. Dataen er ren, konsistent og inneholder ingen mangler. Dette har stor positiv påvirkning på hvor bra modellen vil lære fra datasettet.

For klassifisering av problemer med flere klasser som i vårt tilfelle, blir det anbefalt å bruke 300-500 datapunkt for hver klasse. Ettersom vi har 10 klasser, vil vi egentlig trenge 3000 til 5000 datapunkt. I vårt tilfelle har vi derfor ikke nok datapunkt, i tillegg er noen av klassene underrepresentert. Dette kan heldigvis løses ved resampling teknikker. Og vi kan oversample de underrepresenterte klassene, eller undersample de overrepresenterte. Dette vil gi oss et mer balansert datasett for trening.

MODELLERING

Ettersom dette er et flerklasseklassifiseringsproblem, vil vi bruke klassifiseringsmodeller. Modeller vi utforsker er Randomforest, Support Vector Classifier, Logistic Regression og Extra Trees. Ett tydelig problem med datasettet er at det er ubalansert. Det er underrepresenterte klasser, som vi må resample. Dette kan gjøres ved å ta i bruk SMOTE, fra imblearn og skape ny data for hver klasse. Etter vi har jevnt distribuerte klasser vil vi kunne trene modellene våre. Vi tester ut alle modellene og finner baselinen for hvor bra hver av dem er for å løse problemet. Til

slutt kan det brukes gridsearch sammen med kryssvalidering for å videre finjustere modellene, og det kan være aktuelt å se på feature importance for å eventuelt gjøre endringer i datasettet.

DEPLOYMENT

Det vil bli brukt en web basert løsning. Først Lagrer vi vår endelige modell fra Kaggle, og legger den til i back end. Front end vil bestå av React.js, og back end vil bestå av Flask. front end vil sende en request til back end med brukerens vindata, og Back end vil gjøre en prediksjon med mottatt data med å bruke modellen, og til slutt sende resultatet tilbake til brukeren. For å hoste web applikasjonen bruker vi render.com, som vil kunne hoste både front end og back end.