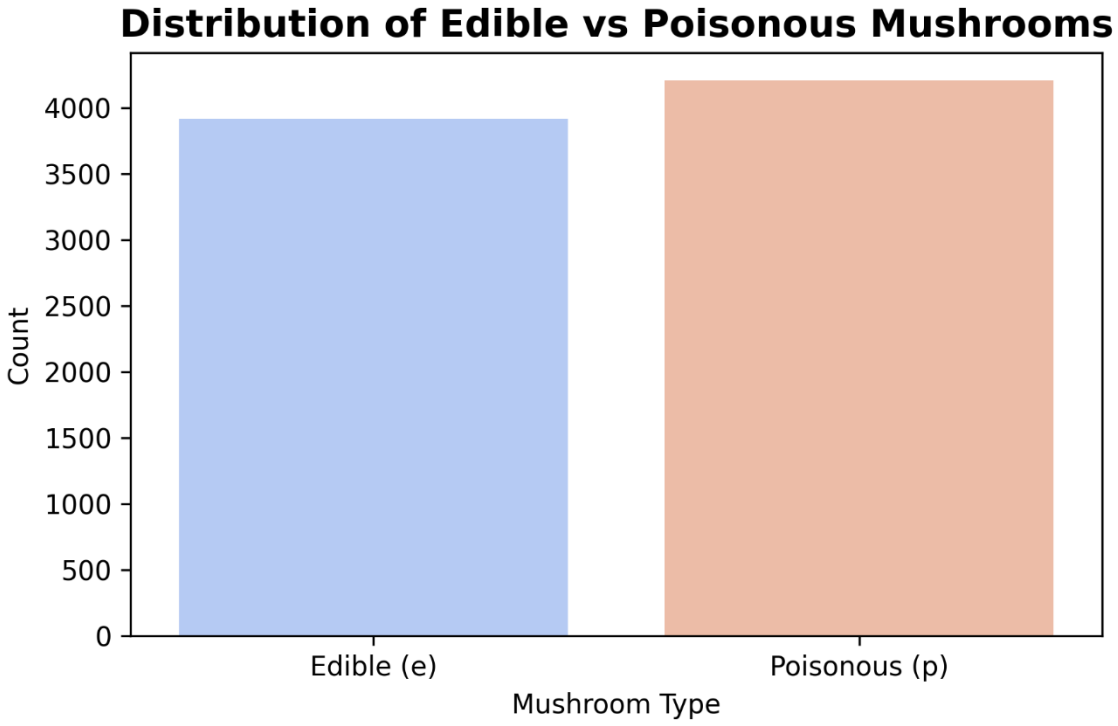
Mushroom Prediction

[Lars Paulsen Løge, Noah Meyer Jamt og Herman Dahlberg], [31.10.2025]

1: OMFANG / SCOPE

Målet med prosjektet er å utvikle en ende-til-ende-løsning som klassifiserer sopp som giftig eller ikke-giftig basert på tabell-egenskaper fra UCI Mushroom-datasettet (f.eks. hattform/-farge, lukt, gjellestørrelse, ringtype og habitat).

Maskinlæring er en egnet løsning fordi datasettet inneholder 8124 observasjoner med 22 ulike egenskaper (features), og det er tilnærmet balanse mellom spiselige og giftige sopper. Dette gjør det mulig å trene en robust modell med lav risiko for bias.



I dag finnes det flere manuelle løsninger på problemet, blant annet visuelle guider og flytdiagrammer for soppidentifikasjon (f.eks. WikiHow). En slik manuell løsning krever at brukeren vurderer hvert trekk individuelt, mens en maskinlæringsmodell kan lære sammensatte mønstre mellom egenskaper som mennesker lett overser.

En ikke-maskinlæringsbasert løsning kunne for eksempel vært en beslutningstabell eller en ekspertregel-basert algoritme. Slike systemer er imidlertid mindre fleksible og vanskeligere å utvide.

Selv om vår modell fungerer svært godt i laboratoriemiljø, er det lite sannsynlig at den vil ha kommersiell suksess. Å bruke en slik modell i praksis innebærer en høy risiko for brukerfeil, og modellen er kun gyldig for «gilled mushrooms» i familiene Agaricus og Lepiota (Dua & Graff, 2019). I tillegg krever løsningen at brukeren legger inn mange egenskaper for å oppnå nøyaktig prediksjon, noe som reduserer brukervennligheten. For å gjøre systemet mer praktisk, benyttes kun de mest informative egenskapene.

Med tanke på Business objective har ikke prosjektet et kommersielt formål, men fungerer som en demonstrasjon på hvordan maskinlæring kan integreres i en enkel webapplikasjon.

METRIKKER

For å evaluere modellen har vi benyttet følgende metrikker:

Accuracy: Måler hvor stor andel av alle prediksjoner som er korrekte. Dette gir et helhetlig bilde av modellens ytelse.

Recall: Måler andelen giftige sopper som korrekt blir klassifisert som giftige. En høy recall er spesielt viktig her for å unngå false positives (at giftige sopper feilaktig klassifiseres som spiselige).

Latency: Er svært lav siden modellen trenes på forhånd og lagres i et pickled objekt. Dette gjør at prediksjoner kan utføres umiddelbart ved bruk.

Siden prosjektet ikke har et kommersielt formål, defineres ingen minimum «business metric»-ytelse. Teknisk sett anser vi prosjektet som vellykket når recall ≥ 0.99.

2: DATA

Datakilde og type:

Vi bruker UCI Mushroom-datasettet (ID 73). Det består av ca. 8124 observasjoner og 22 kategorier.

Tilgang og innsamling:
Data hentes via `ucimlrepo.fetch_ucirepo(id=73)` i treningsskriptet.

Datamengde nå og behov.
Hele datasettet er tilgjengelig og tilstrekkelig for tren/test og kryssvalidering for denne oppgaven. Siden alle trekk er kategoriske og lav-dimensjonale etter one-hot encoding, kreves det ikke mer data for å nå høy ytelse på denne oppgaven.

Labels og konsistens.
Label følger UCI-definisjonen og er allerede oppgitt i datasettet. Konsistens sikres ved å:

- Laste labelkolonnen uendret,
- Bruke stratifisert tren/test-splitt (for å bevare klasseforholdet),
- Sette faste `random_state`-verdier for reproduserbarhet.

Etikk og personvern.

- Datasettet inneholder ingen personopplysninger.
- Ansvarlig bruk: Modellen skal ikke faktisk brukes som et verktøy for å bestemme om en sopp er spiselig eller ikke, dette prosjektet er bare brukt som et eksempel på hvordan man kan lage en nettside applikasjon som bruker maskinlæring.

Representasjon og forbehandling.

- Alle trekk er kategoriske og representeres som én bokstavkode.
- Vi benytter One-Hot-Encoding via `OneHotEncoder(handle_unknown='ignore')` inne i en `ColumnTransformer`.
    - `fit` kjøres kun på treningssettet; `transform` på val/test og i appen.
    - `handle_unknown='ignore'` gjør løsningen bedre mot sjeldne/ukjente verdier (f.eks. manglende verdier i stalk-root featuren).
- Skalering er ikke nødvendig (alt er diskret/kategorisk).
- Rensing/feature engineering:

    - Vi behandler '?' som egen kategori (ingen imputasjon).
    - Ingen ekstra feature engineering er nødvendig.
- Preprosessor (encoder) og modell lagres separat som `mushroom_preprocessor.pkl` og `mushroom_model.pkl`, som appen laster ved oppstart. Dette matcher kodebasen.

Observasjon om 100 % accuracy:
Flere modeller (inkludert vår) oppnår 100 % nøyaktighet på dette datasettet. Dette er plausibelt fordi datasettet er nesten perfekt separerbart, spesielt på grunn av feature *odor*.

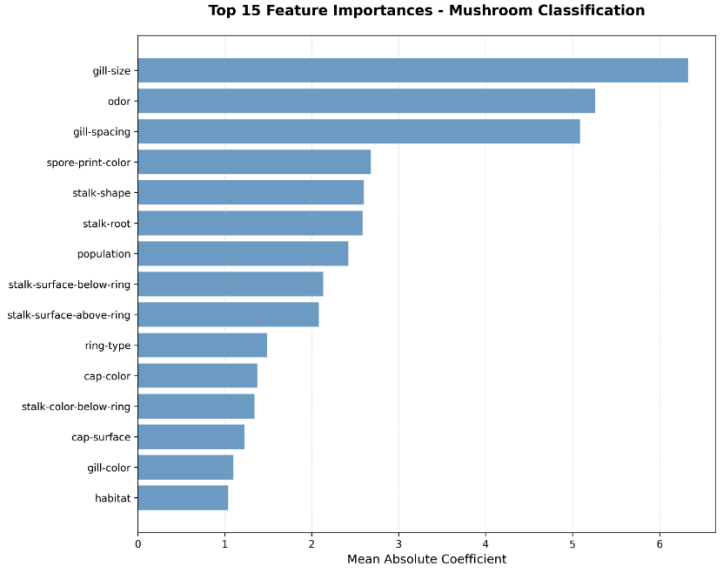
For økt brukervennlighet valgte vi å redusere antall features til de seks mest informative: 'gill-size', 'odor', 'gill-spacing', 'stalk-surface-above-ring', 'spore-print-color', og 'stalk-root'. Dette reduserte accuracy fra 1.00 til 0.997, noe vi anser som akseptabelt.

3: MODELLERING

Vi brukte stochastic gradient decent (SGD). Random forest trees ble også testet, men de hadde akkurat samme accuracy score på 1.00 når vi brukte alle features. SGD fikk som nevnt tidligere en accuracy score på 0.9970 med kun de viktigste features.

En baseline uten maskinlæring (f.eks. tilfeldig gjetning) vil gi ca. 50 % nøyaktighet siden datasettet er balansert.

Ved hjelp av coef_-verdiene fra modellen analyserte vi hvilke features som hadde størst innvirkning. Disse innsiktene ble brukt til å fjerne uviktige trekk og forbedre brukervennligheten uten å ofre ytelse. Vi valgte de seks største features å trene modellen på og gjøre prediksjoner på input fra brukere. 'gill-size', 'odor', 'gill-spacing', 'stalk-surface-above-ring', 'spore-print-color' og 'stalk-root'.



På nettsiden vi fant datasettet var det også baseline accuracy og precision scores fra andre modeller som Xgboost, Support Vector og Random Forest Classification. Alle disse oppnådde 100.0.

4: DEPLOYMENT

Modellen er deployert som en webapplikasjon ved hjelp av Gradio (Abid et al., 2019), og hostes på Hugging Face Spaces (Hugging Face, 2023).

Applikasjonen laster inn mushroom_preprocessor.pkl og mushroom_model.pkl, tar imot brukerens valg av sopp-egenskaper via nedtrekksmenyer, og returnerer både klasselabel og sannsynlighet for giftighet.

Ved å bruke Hugging Face Spaces får man en skybasert og gratis plattform for å distribuere maskinlæringsapplikasjoner med minimal konfigurasjon. Gradio håndterer frontend og backend i ett, slik at brukeren får et responsivt grensesnitt uten behov for ekstra web-rammeverk.

Systemet krever ikke regelmessig vedlikehold, ettersom soppens egenskaper ikke endres over tid. Det er heller ikke behov for retrening av modellen.

5: REFERANSER

- *How to Identify Edible Mushrooms.* WikiHow. Hentet fra https://www.wikihow.com/Identify-Edible-Mushrooms
- Dua, D. & Graff, C. (2019). *UCI Machine Learning Repository – Mushroom Dataset.* University of California, Irvine.
  Abid, A., Abdalla, A., m.fl. (2019). *Gradio: Hassle-Free Sharing and Testing of MLModels in the Wild*. (Dokumentasjon/bruk til web-app.)
- Hugging Face. (2023). *Hugging Face Spaces: Deploy Machine Learning Apps.* Hentet fra https://huggingface.co/spaces

6: BRUK AV AI-VERKTØY

Vi har brukt ChatGPT som verktøy til feilsøking i koden og til hjelp med ordlegging i rapporten.