Titanic Assignment

Machine Learning

A) Getting to know the data and the problem.

Problemstillingen i Titanic opgaven er at lave en model som kan forudsige om en given passager ombord overlevede eller døde, ud fra datasættet. Det er altså et binær klassifikationsproblem, da resultatet som der forudsiges enten er: overlevet(1) eller død(0). For at danne denne forudsigelse, skal datasættet analyseres ved at bruge diverse ML-algoritmer. Disse algoritmer finder mønstre og sammenhænge mellem datapunkterne ved at kigge på de tilhørende attributter for hver passager. Ved at undersøge denne data kan der bygges en model som kan forudsige nogenlunde præcist om en given passager overlevede eller døde under Titanic.

De forskellige data der indgår i datasættet er:

1. Overlevet
2. Klasse
3. Navn
4. ID
5. Køn
6. Alder
7. Antal Søskende ombord
8. Antal forældre ombord
9. Billetnummer
10. Billetpris
11. Kahytnummer
12. Indskibningssted

B) Cleaning the data.

Ud fra disse 12 forskellige datapunkter for hver passager kan man finde frem til, om passageren overlevede. Datapunktet 'Overlevet' er i dette tilfælde den værdi som skal forudsiges, så den bruges til at krydsreferere resultatet til sidst, for at se hvor præcis modellen er. Punktet svarer altså til vores Y-data.

Data såsom navn, id, billet- og kahytnummer kan i dette tilfælde blive slettet fra listen, eftersom datakvaliteten for disse enten ikke er fuldendte samt ligegyldige for modellen.

Nogle datapunkter i de resterende 7 mangler også noget data, men det kan der rettes op på. Fx mangler nogle passagerer deres alder i datasættet, hvilket er et vigtigt datapunkt for forudsigelsen af overlevet. Den brugte teknik til at tilregne en fornuftig alder for hver passager uden en alder, er at bruge en ML-

algoritme der forudsiger den givne passagers alder ud fra de tilhørende datapunkter. Der trænes altså en regressions-model til at definere en passende alder i stedet for at tage et tilfældigt tal eller gennemsnittet.

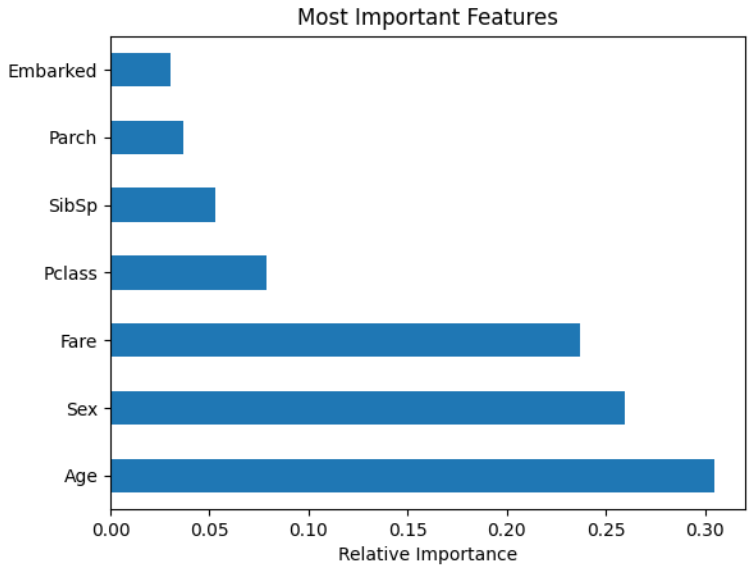
Samtidig med at alderen for hver passager bliver genereret, bliver deres køn omdannet da datatypen består af en streng, altså ikke-numerisk data. Denne data bliver konvertereret til enten 0 eller 1 alt efter hvilket køn passageren har. Dette er nødvendigt for at ML-algoritmen kan processere og klassificere de forskellige data. Der bliver brugt label-encoding til at lave denne konvertering.

Efter dataene er renset og processeret, splittes det i træning- og test-sæt. Træningssættet bliver brugt til at træne ML-modellen, mens testsættet bruges til at evaluere modellens ydeevne og præcision. Der er brugt den gængse 80/20-størrelse af de to sæt. Træningssættet består altså af 80% og testsættet for de resterende 20%. Da der kun er 800 datapunkter i datasættet, kunne man argumentere for en større andel til at træne modellen.

C) Choosing a model and doing training.

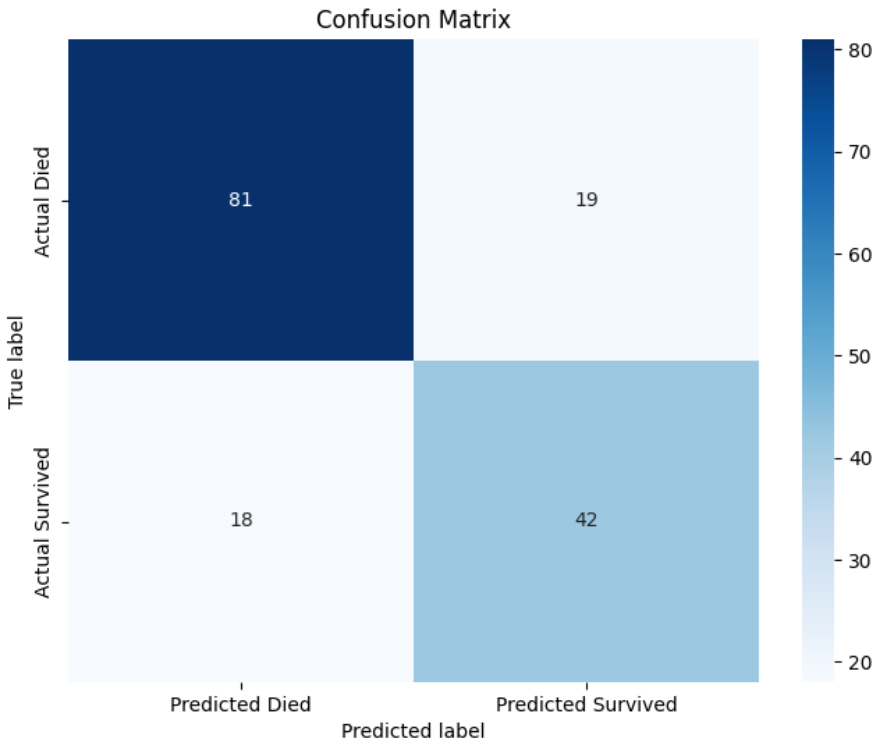
Da dataene nu er klar til brug i diverse algoritmer, skal der vælges en model som benytter dataene til at skabe en model, der bedst muligt kan forudsige om en given passager overlever. I denne opgave bliver Random Forest Classifier brugt som model. RF bruger mange decision trees når modellen trænes, og laver en regression over disse, til bedre at kunne forudsige det rigtige udfald. Yderligere er RF oftest robuste og overfitter sjældent sammenlignet med andre modeller. Denne robusthed gør RF til et godt valg når man har datasæt som er komplekse relationer mellem data eller med meget støj i datakvaliteten.

Der findes en indbygget funktionalitet i RF som kigger på hvilke data har størst betydning for modellen. Det giver et indblik i, hvilke faktorer der er mest betydningsfulde for overlevelse på Titanic. Samtidig belyser det potentielle mønstre i sin data og kan belyse hvilke data der måske er overflødig for modellen.



D) Evaluating performance on the test set.

Til at beregne hvor god modellen er, kan vi beregne precision, recall, accuracy og confusion matrix. Precision og recall er vigtige at kigge på, for at tegne et billede af hvor præcis modellen er for Titanic datasættet. Precision fortæller os hvor mange forventede positive forudsigelser faktisk er positive. En høj precision indikerer få falske positive. Recall fortæller os hvor mange faktiske positive udfald der var forudsagt korrekt af modellen. En høj recall indikerer færre falske negative. Confusion matrix giver en opdeling af korrekte og ukorrekte forudsigelser for modellen. Accuracy viser hvor stor en andel forudsigelser der var korrekte i testsættet.



	precision	recall	f1-score	support
died	0.82	0.81	0.81	100
survived	0.69	0.70	0.69	60
accuracy			0.77	160
macro avg	0.75	0.76	0.75	160
weighted avg	0.77	0.77	0.77	160

I Confusion Matrixen kan man se antallet af True Positives (42), True Negatives (81), False Positives (19), False Negatives (18). Derudover kan man se at modellens precision når den skulle forudsige om man døde er 82% og modsat 69% for at overleve. Den er altså markant bedre til at forudsige om en given passager dør.

På accuracy-scoren kan man se hvor korrekt modellen er. Den scorer 77% overordnet set, hvilket betyder at 77% af alle forudsigelser modellen har lavet på testsættet, var korrekte. Eftersom 20% af datasættet bliver brugt som testsæt, er der her 160 passagerer som modellen gætter på.

E) Experiments

Hvis man kigger på grafen over de vigtigste features for modellen, kan man se de tre øverste ikke har lige så stor betydning som de fire nederste. Hvis man fjerner dem fra datasættet og ændrer på forholdet mellem træning- og testsæt til 90/10 i stedet for, ender modellen med at yde således:

	precision	recall	f1-score	support
died	0.85	0.81	0.83	48
survived	0.74	0.78	0.76	32
accuracy			0.80	80
macro avg	0.79	0.80	0.79	80
weighted avg	0.80	0.80	0.80	80

Accuracy for modellen er nu 80% vs. 77% fra før, samtidig med at præcisionen for at forudsige om en person overlever, er steget markant. Det tyder på at hvis man fjerne det unødvendige data samt øger træningssættet, kan modellen bedre forudsige udfaldet for hver passager ombord Titanic.