TLOG2011 – Prosjekt 2

Introduksjon

I dette prosjektet har vi laget en maskinlæringsmodell som skal bli brukt til å predikere om en matleveranse vil være forsinket eller ikke. Datasettet vi har brukt i dette prosjektet er gitt ut av Meituan, et kinesisk selskap som driver med matlevering. Målet med å kunne predikere om en leveranse vil være forsinket er for å kunne yte bedre service til sluttkunden, slik at ikke de forventer mat tidligere enn de får den.

Fremgang

For å kunne løse dette problemet måtte vi først rense datasettet og dele det opp i et treningssett og et testsett. Dette gjorde vi i Python ved hjelp av Pandas metoder. Deretter måtte vi finne ut hvilken metode modellen vår skulle bruke for å kunne løse problemet. Vi forsøkte først med logistisk regresjon og random tree, men disse ble for enkle for problemet. Vi endte opp med å bruke XGBoost, da denne kan finne mer komplekse sammenhenger enn enklere modeller.

Et problem vi hadde var at datasettet var skjevfordelt, hvor ca. 95% av datasettet var «on-time» ordrer, mens bare 5% var «late». Dette førte til at de enklere modellene klarte å oppnå en høy nøyaktighet, men dårlig presisjon og recall. XGB fikk noe lavere nøyaktighet mot at vi fikk mye bedre presisjon og recall.

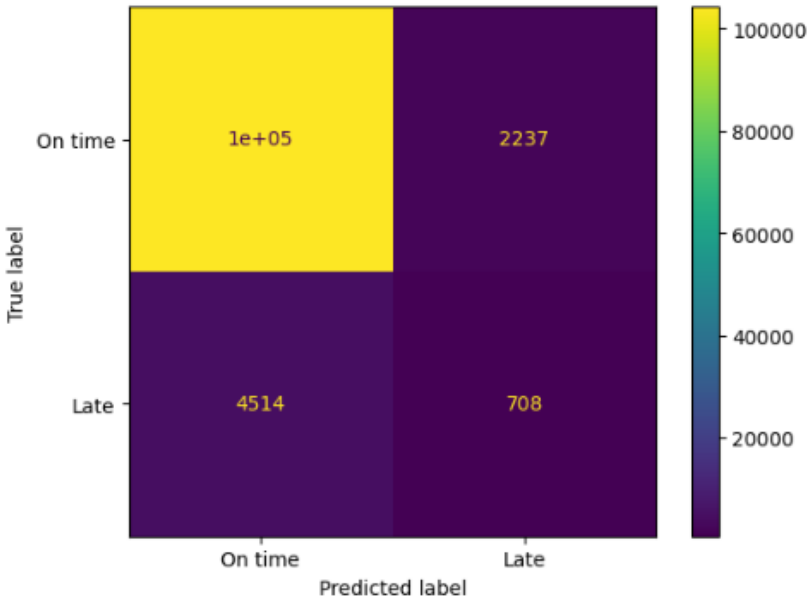
XGBoost

Når vi brukte XGBoost metoden endte vi opp med confusion matrixen under, hvor vi ser at modellen klassifiserte rundt 100 000 riktig på on time (true negative) og 708 riktig på late (true positive). Samtidig lager den en del falske alarmer med 4514 sene leveranser som klassifiserer som tidsnok, og 2237 on time leveranser som klassifiseres som forsinket.

Modellen oppnår

- Nøyaktighet: 93.96%
- Presisjon: 24.04%
- Sensitivitet: 13.56%

Presisjon på rundt 24% vil si at nesten 1 av 4 ordre som klassifiseres som forsinket faktisk er forsinket. Når vi først brukte XGBoost slet vi med at denne verdien var veldig lav (rundt 9%), og for å øke den måtte vi øke treshholden til modellen til 0.7. Dette medførte da at nøyaktighet gikk litt opp og sensitivitet gikk litt ned, noe vi mener er et greit kompromiss.

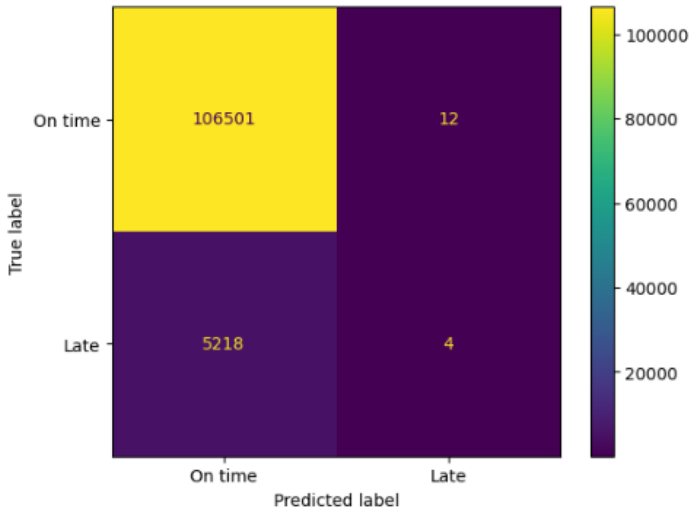


Logistisk regresjon

Logistisk regresjon var veldig nøyaktig for oss, da de nesten bare predikerte ordrene som on time. Bare 16 av over 110 000 ordre ble klassifisert som forsinket, hvor bare 4 var riktig. Med så få riktige sene leveranser får man veldig lav sensitivitet.

Modellen oppnår

- Nøyaktighet: 95.32%
- Presisjon: 25%
- Sensitivitet: 0.08%

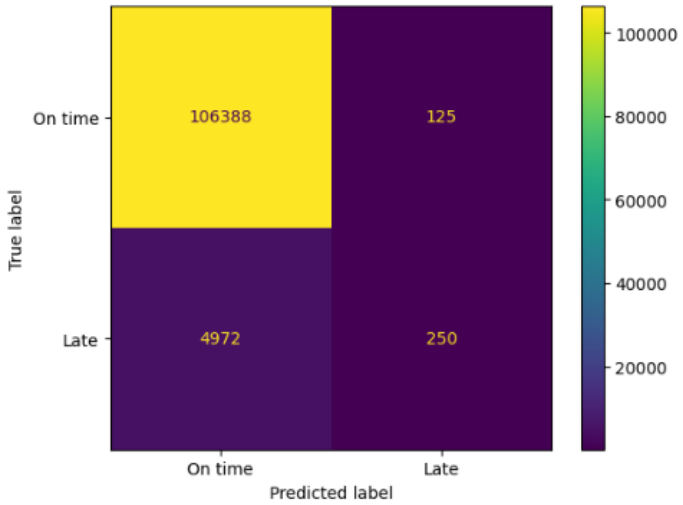


Random forest

Random forest ga oss høy nøyaktighet og veldig bra presisjon, men også denne metoden ble for konservativ og endte med lav sensitivitet. Denne modellen klassifiserte omtrent like mange riktige «on time» leveranser, men fikk også en del mer riktige «late» leveranser. Denne modellen er bedre enn logistisk regresjon, men ikke like bra som XGBoost.

Modellen oppnår

- Nøyaktighet: 95.44%
- Presisjon: 66.67%
- Sensitivitet: 4.79%



Sammenligning

Modell	Nøyaktighet	Presisjon	Sensitivitet
XGBoost	93.96%	24.04%	13.56%
Logistisk regresjon	95.32%	25%	0.08%
Random forest	95.44%	66.67%	4.79%

Konklusjon

Vi endte som sagt med å bruke XGBoost som metode for vår modell. Her kunne vi også forsøkt å bruke andre metoder som Catboost for å se om vi fikk bedre resultater. Alle 3 modellene ga til slutt omtrent lik nøyaktighet (ca. 1.5% forskjell fra lavest til høyest), men XGBoost hadde mye høyere sensitivitet sammenlignet med de andre. I en logistikkontekst kan en modell med høy presisjon redusere unødvendige tiltak, men lav sensitivitet innebærer at mange faktiske forsinkelser ikke fanges opp. XGBoost gir den beste balansen mellom disse hensynene.