

## **ML – Housing Prices prediction**

**Øyvind Onarheim, Tor Magne Solheimsnes og Andre Ripman**

**15.11.23**

### **SCOPE**

Målet med dette prosjektet er å ha en pålitelig og god løsning for å gi pris til eiendommer, enten om det er slik at privatpersoner kan bruke dette eller at megler firma kan verifisere e-takster eller bruke dette som en egen modell for prissetting. Per i dag, om en privatperson skal eksempelvis søke om refinansiering av lån, så kan de enten skaffe en taksering av eiendom eller de kan bruke pris på bolig som ligger i et boligregister for å vite hvilke verdi de kan stille med i pant. Dette registeret tar ikke hensyn til påbygg, utsikt, eller individuelle løsninger på den eiendommen, men går heller ut ifra snitt verdi i nabolaget og hva tilsvarende hus i nærheten har blitt solgt for. Denne maskinlærings modellen vil gi en langt mer individuell og korrekt pris.

Man kan måle ytelsen til modellen ved å se på eventuelle kostnader spart for enten kunde eller meglerfirma/bank. For banken sin del vil dette resultere i en mer korrekt pant, der det er viktig at bolig ikke blir overestimert i pris dersom pant må innkreves for å dekke lån. Stakeholders i dette prosjektet vil være bank og/eller meglerfirma.

Tidslinje:

### **Problemdefinisjon (Uke 1-2)**

- Definere formålet med prosjektet
- Analysere datasett og se hvilke features man har å jobbe med

### **Datainnsamling og forberedelse (Uke 3-6)**

- Samle inn all relevant data om bolig, inkludert alle relevante features for datasettet.
- Klargjøre data for bruk i implementasjon

### **Modellvalg og implementasjon (Uke 7-10)**

- Finne riktig maskinlæringsalgoritme for regressjon.
- Implementere og trene modell

### **Evaluerings (Uke 11-12)**

- Evaluere modellens ytelse
- Finne forbedringer
- Finne viktigste features

### **Dokumentasjon og implementasjon (Uke 13-16)**

- Dokumentere alt om prosjektet og modellen, og om prosessen.
- Integrer modellen i produksjon

### **Oppfølging og vedlikehold (etter uke 16)**

- Kontinuerlig overvåk modellen sin ytelse, og oppdater modell etter behov

I et slikt prosjekt må man ha en del ressurser og man trenger investeringer innenfor mange forskjellige områder for å få et optimalt resultat. Eksempler på dette er:

- Datasett innsamling

- Maskinvare. Man trenger en kraftig datamaskin eller server som skal kunne kjøre denne modellen, og man må beregne å måtte utvide dette etter som flere og flere bruker modellen
- Verktøy for å utvikle modellen. Tensorflow, Pandas, etc
- Fagkompetanse. Man trenger eksperter innenfor både maskinlæring, dataanalyse, økonomer, statistikere og domeneeksperter.
- Testing og kvalitetssikring

## DATA

I dette prosjektet bruker vi data som er samlet inn om alle eiendommer i et gitt område. Men når dette prosjektet skal i produksjon, skal man kunne prediktere prisene på boliger i hele Norge. Dette er data som kan samles inn via meglerfirmaer og banker, da de har register over veldig mange eiendommer som er solgt i Norge, samt man kan bruke eiendomsregisteret for å hente ut diverse informasjon om boliger. Da stakeholders i dette prosjektet er banker og eiendomsmeglere, kan de hente inn nødvendig data om eiendommer slik at det samsvarer med modellens behov. Vi estimerer at vi trenger nok data til å få et eksakt bilde på korrekt pris, uansett hvor i landet eiendommen ligger. Dette vil si at vi trenger data fra alle byer og områder i Norge, der dette er tilgjengelig. Vi gir en oppskrift til bank/megler om hvordan vi ønsker at data skal samles inn, og i hvilket format, som gjør at vi har korrekte labels og features slik som modellen krever det.

Eiendomsdata, spesielt knyttet mot boligpriser, er sensitiv informasjon. Så vi må sørge for at all datainnsamling, lagring og behandling av data er i samsvar med alt av personvernlover som eksempelvis GDPR og man må ha tilstrekkelig med sikkerhetsrutiner rundt modellen for å hindre at uvedkommende får tilgang på dataen. Man må også i noen tilfeller kanskje måtte hente inn samtykke fra eier, og data må anonymiseres. Etske retningslinjer kan være at modellen ikke må diskriminere basert på faktorer som ligger utenfor de variablene modellen skal bruke, som eksempelvis informasjon om eier av bolig, etc.

Det er nødvendig å håndtere manglende verdier, feature engineering, skalere og normalisere dataen samt å sørge for at data er anonymisert gjennom hele prosessen.

## MODELLERING

Når vi deltok i Kaggle-konkurransen med fokus på boligpriser, startet vi med å utforske en rekke maskinlæringsmodeller. Vi begynte med det grunnleggende, som enkel lineær regresjon, for å etablere en forståelse av forholdet mellom de ulike funksjonene og målvariabelen, i dette tilfellet salgspris. Dette ga oss en base for å sammenligne fremtidige, mer komplekse modeller mot. Videre brukte vi metoder som TensorFlow Random Forest, som er dyktige til å håndtere et stort antall funksjoner og redusere risikoen for overfitting, en vanlig utfordring i maskinlæring. Videre, utforsket vi regresjonsteknikker som Gradient Boosting, for eksempel gjennom implementeringer som XGBoost.

Estimering av baseline-ytelse er avgjørende for å sette en standard for forventet modellnøyaktighet. Enkel statistikk, som å beregne gjennomsnitt eller median av salgspriser, kan tjene som et utgangspunkt. Å sammenligne med tidligere arbeid og resultater oppnådd av andre i Kaggle-konkurransen ga oss verdifull innsikt.


Modellering i maskinlæring er en iterativ prosess. Basert på innsikt fra flere tester av modellen, har vi justert og forbedret den kontinuerlig.


## DEPLOYMENT

For å sette modellen i drift og bruke prediksjonene, må man iverksette modellen i produksjonsmiljøet, via enten API eller lignende. Etter modellen er satt i drift må man kontinuerlig overvåke ytelsen, og se at modellen ikke degradere i ytelse og at man holder den oppdatert med ny data. Det må også planlegges for eventuelle feil i modellen og hva som skal gjøres dersom man oppdager at modellen har gitt feil informasjon. Som en del av dette, burde man ta kontinuerlig tilbakemeldinger fra brukere av systemet, for å gjøre nødvendige tilpasninger.

## REFERANSER

Brukt ChatGPT til forslag på oppsett av rapport og kildeliste. Også brukt ChatGPT til å gi beskrivelse av noen linjer, som eksempelvis hva en tentativ tidslinje er.

 **You**  
Hva er en tentativ tidslinje

 **ChatGPT**  
En tentativ tidslinje er en foreløpig plan eller tidsplan som gir et overordnet bilde av når ulike aktiviteter eller milepæler forventes å skje i løpet av et prosjekt. Den hjelper med å strukturere arbeidet og gir et rammeverk for å styre progresjonen. En tentativ tidslinje er ofte justerbar etter hvert som prosjektet utvikler seg, og den gir et grovt estimat over tidsrammene basert på tilgjengelig informasjon og ressurser.

Brukt GitHub copilot for forslag til hvordan vi skal konvertere objekter til numeriske verdier, og integrere de i en ny liste som er sortert likt som test data filen. Vi opprettet først medianen for alle verdiene i datasettet, men fant ut at dette var en lite optimal måte å gjøre det på, og ba copilot om å endre til en for loop.