Refleksjonsnotat

Datainnsamling, databehandling, dataanalyse og visualisering

Gjennom dette prosjektet har vi fått et helhetlig innblikk i prosessen med å hente inn, strukturere og analysere miljødata fra åpne kilder. Vi jobbet med historiske værdata hentet fra SeKlima og Yr, noe som ga innsikt i hvordan åpne datakilder kan benyttes til faktabaserte analyser. Vi erfarte hvor viktig det er å forstå datastrukturen før man går videre med behandling og analyse.

Databehandlingen innebar blant annet å rydde opp i feilformatert data, konvertere tidssoner og gjøre dataen klar for videre analyse. Deretter utførte vi statistiske analyser og utviklet enkle regresjonsmodeller for å avdekke mønstre i temperatur- og vinddata. Til slutt visualiserte vi funnene med hjelp av interaktive grafer for å gjøre innsikten mer tilgjengelig.

Nye ferdigheter og verktøy

Vi har blitt mer komfortabel med flere sentrale biblioteker i Python som Pandas, Numpy og Matplotlib. Vi har fått en god forståelse for hvordan en kan behandle store mengder med data ved bruk av pandas og blitt godt kjent med hvordan man skal lage fine og oversiktlige grafer med matplotlib og seaborn. Vi har i tillegg fått bruk for pandasql noe ingen av oss visste eksisterte før dette prosjektet.

Samtidig som vi ble mer komfortabel med de mest sentrale bibliotekene har vi også lært oss å bruke mer kompliserte biblioteker som sklearn, plotly og unittests funksjon patch. Sklearn gjorde beregningen av regresjonslinjen veldig enkel. Plotly var oversiktlig og enkelt å bli kjent med. Samtidig så var bruken av patch funksjopnen helt nytt for oss da vi skulle lage tester, men den er genial da en skal late som funksjonen gjør noe annet enn den egentlig gjør. I tillegg til teknisk kunnskap har vi forbedret vår forståelse av hvordan man dokumenterer og strukturerer et datavitenskapsprosjekt på en ryddig måte da har vi fått kjennskap til en mer objektorientert framgangsmåte innen programmering.

Utfordringer

En av hovedutfordringene var å sette opp det virtuelle miljøet. Dette brukte vi lang tid på da alt av oppskrifter og hjelpemidler (chatGPT, StackOverflow, osv.) ikke fungerte, vi endte opp med å søke hjelp hos faglærer da vi brukte i overkant av 45 minutter før vi endelig ordnet problemet.

Ellers møtte vi et problem da vi skulle hente samme data som orginalt var hentet med Seklima ved bruk av frostapi. Problemet der lå i at det ikke eksisterte data for høyeste vindkast per time ved Slettnes fyr. Dermed endte vi opp med å droppe bruken av API totalt da det heller ikke var nødvendig for historisk data.

Gruppens samarbeid

Gruppens samarbeid fungerte godt og kommunikasjonen foregikk effektivt via felles chat.

Vi kunne med fordel hatt en mer strukturert arbeidsmetodikk da «pushing» inn i GIT repository var relativt uoversiktlig..

Vurdering av resultatene

Resultatene vi oppnådde var solide. Visualiseringene ga god innsikt i værmønstre og de statistiske analysene var forståelige og relevante. Vi klarte å levere et gjennomarbeidet prosjekt med god kodedisiplin, ren struktur og fungerende databehandling.

Visualiseringene gjør det enkelt å tyde trender som generelt mindre vind ved høyere temperaturer (lett å se i heatmapene) og å se hvilke verdier som forekommer mest.

Forbedringspotensial

- Prosjektledelse: Mer systematisk bruk av versjonskontroll og oppgavefordeling.
- Dokumentasjon: Tydeligere kommentarer i koden og brukerveiledninger for hvordan analysene skal kjøres.
- Datamengde: Skulle hatt mer data og jobbe med for å kunne utføre mer grundige analyser.
- Bruk av API: Skulle brukt API-er for å kunne hente inn data, slik dataen lettere kunne vært hentet frem til dagens dato. Da ville vi kunne sjekket egen prognose opp mot prognoser fra åpne kilder.
- Prognoser: Lineær regresjon i tidsplanet er ikke en veldig god måte å predikere verdier frem i tid. Det ville passet bedre med mer sykliske regresjonslinjer som sinus, eller logistiske funksjoner.
- Requirements.txt: Requirements.txt filen inneholder alt av biblioteker som er tatt i bruk og inkluderer også underbibliotekene som brukes. Det gjør at den er veldig uoversiktlig.
- Logging: Skulle helst tatt i bruk logging av data, altså at alt av hendelser blir registrert som hjelper med å overvåke programflyt og feilsøke problemer.

Veien videre

Prosjektet åpner for mange mulige utvidelser. Et naturlig neste steg kunne være å integrere et API for å hente sanntidsdata og sammenligne historiske trender med nåsituasjonen. Vi kunne også brukt maskinlæring for å predikere fremtidige værforhold basert på tidligere data på andre måter enn lineært.

Viktige læringspunkter

- Verdien av godt strukturerte og rensede data
- Samspillet mellom kode, visualisering og statistikk
- Hvordan samarbeidsverktøy og versjonskontroll styrker et gruppeprosjekt
- Økt forståelse for hvordan miljødata kan brukes i praktiske, samfunnsnyttige analyser

Personlige refleksjoner

Prosjektet har gitt oss en konkret opplevelse av hvordan teori fra datavitenskap og miljøstudier anvendes i praksis. Erfaringen vi har fått her vil være verdifull i videre studier og i fremtidig arbeidsliv – spesielt innen maskinlæring da vi skal ha faget «TDT4172» neste semester. Ellers har det gitt erfaring innen analyse, datadrevet beslutningstaking og utvikling av digitale løsninger.