Norges miljø- og biovitenskapelige universitet

Masteroppgave 2024 30 stp
Fakultet for Realfag og Teknologi

Evaluering av semi-veiledede algoritmer for regresjon og bruk av spektroskopisk data til optimalisering av biokjemisk prosessindustri

Evaluating Semi-Supervised Regression Algorithms and Use of Spectroscopic Data for Optimizing Biochemical Processes

Pradeep Manoraj og Trishaban Jegatheeswaran
Industriell Økonomi og Datavitenskap

Blank Side

Forord

Denne masteroppgaven markerer slutten på vår reise etter alle våre år som har blitt tilbrakt ved Norges miljø- og biovitenskapelige universitet (NMBU) på Ås, Norge. Det har vært en lang og lærerik reise, med både gode og harde tider som har satt grunnlaget for veien videre i livet.

Vi ønsker å uttrykke vår takknemlighet til våre flinke veiledere. Hovedveileder Prof. Dr. Kristian Hovde Liland har gitt oss et lærerikt innblikk i maskinlærings-verden med sine vise ord som professor på universitetet og bidratt med interessante idéer til våre utfordringer utover masterperioden vår. Biveileder Dr. Ingrid Måge fra Nofima har vært en utrolig sterk ressurs, som har gitt oss både verdifull veiledning og utallige tilbakemeldinger, etter mange år med fremragende forskning i matindustrien. Videre ønsker vi å takke Nofima for et omfattende og lærerikt prosjekt og en fin arbeidsplass. Deretter ønsker vi å takke Bioco som ga oss muligheten til å jobbe med reell data fra prosessindustrien. Vi ønsker å takke overingeniør Uzair Aftab fra EIK-LAB som tilbød deres super-PC og andre ressurser for å akselerere behandlingstiden til koden vår.

Til slutt ønsker vi å takke vår kjære familie og venner som har støttet oss og delt inspirerende ord utover reisen vår. Videre ønsker vi å si en takk til kollektivet og alle andre medstudenter på campus som har skapt minner for livet.

Ås, 15. Juni, 2024

Pradeep Manoraj **Trishaban Jegatheeswaran**

Blank Side

Sammendrag

Maskinlæring har muligheten til å finne banebrytende løsninger som kan effektivisere ulike sektorer ved å utnytte tilgjengelig data. I en biokjemisk prosessindustri blir data målt kontinuerlig. Data som måles er typiske kontrollparametere som sørger for at prosessen foregår som planlagt. Ved å utnytte et slikt sett med data er det potensiale for å anvende maskinlæringsalgoritmer til å optimalisere stegene i prosessen. Likevel er det både tidkrevende og kostbart å samle inn kvalitetsmålinger. Datapunktene med disse målingene kalles for markerte data, mens de uten kalles umarkerte data. Når det er få kvalitetsmålinger, hindrer det klassiske veiledede maskinlæringsalgoritmene i å prestere optimalt. Følgende problem har ført til en økt interesse for semi-veiledede maskinlæringsalgoritmer. Fordelen med semi-veiledet læring (SSL) er at det er muligheter for å benytte seg av både umarkerte og markerte data, i kontrast til klassisk veiledet maskinlæring (VL) som begrenser seg til kun markerte data.

Denne forskningen går ut å evaluere mulighetene av å bruke semi-veiledede maskinlæringsalgoritmer for å predikere produktkvaliteten til råmaterialet som blir behandlet i den biokjemiske prosessindustrien. Oppgaven tar utgangspunkt i problemstillingen om å predikere produktkvalitet ved bruk av kvalitetsmålinger med semi-veiledede algoritmer og spektroskopisk data. Undersøkelsene gikk ut på å anvende tre ulike semi-veiledede algoritmer. Deretter ble de sammenliknet med tre klassiske veiledede maskinlæringsalgoritmer. For å sammenlikne algoritmene ble de vurdert etter ulike evalueringsmetrikker for regresjonsproblemer innen maskinlæring. Følgende metrikker som ble benyttet er RMSE, R^2, MAE og MAPE. Metrikkene ble brukt for å skille grad av nøyaktighet mellom modellene. Deretter ble en forenklet MCDA analyse innført for å vurdere de praktiske aspektene for mulig implementering.

Formålet med studien er å gi innsikt til modeller som kan ta i bruk umarkerte data, i tillegg til spektroskopiske data. Dersom modellene har optimal ytelse, kan det bidra til å forbedre dagens situasjon. Potensialet er spesielt relevant når det kommer til beslutningstakinger om håndtering av produkter basert på produktkvalitet.

Studien er knyttet til et virkelig datasett av en bioprosess, hos bioraffineriet Bioco AS. I prosessen blir råmateriale fra både kylling og kalkun behandlet. Det ble testet for ulike enzymtyper i prosessen, og i denne fasen ble det samlet data av de tilgjengelige måleinstrumentene. I tillegg til de tradisjonelle instrumentene ble det samlet spektroskopisk data, ved hjelp av en midlertidig utplassert NIR-sensor.

Under innsamlingen av data ble det utført to forskjellige produksjoner etter hverandre, hos Bioco. Den første produksjonen gikk ut på teste ulike enzymtyper på råmaterialet. Dette var en testfase for å se hvordan det påvirker den behandlede massen. Den andre produksjonen var å gå tilbake til standardproduksjonen hos bedriften. Det baserer seg på å bruke én fast enzymtype. Da kvalitetsmålingene ble samlet inn var det hovedsakelig prioritert å samle inn ulike målinger fra forsøksperioden, nemlig den første produksjonen. Dette fører til at det er en ujevn fordeling av markerte og umarkerte data i datasettet, fordi kvalitetsmålingene ble tatt med hensyn til de ulike enzymtypene. En konsekvens av

dette er at det fører til en større konsentrasjon av den faste enzymtypen fra standardproduksjonen i den umarkerte andelen av datasettet, i motsetning til de ulike enzymtypene fra forsøksperioden.

Våre funn viser til at veiledede algoritmer presterer bedre enn semi-veiledede algoritmer, til tross for at den førstnevnte metoden er begrenset med kun markerte data. Semi-veiledede algoritmer baserer seg på store deler av umarkerte data. Det ble videre vist at det var klare forskjeller mellom algoritmene når det ble brukt spektroskopisk data i treningsfasen.

MCDA-analysen viser til at Alternativ 2 blir sett på som den ideelle løsningen. På grunn av konfidensialitet, var økonomisk data rundt sensor og innsamling av kvalitetsmålinger utilgjengelig. Dermed er det viktig å være klar over at valgene tatt i denne analysen er basert på den spesifikke prosessen til Bioco. Denne anbefalingen er begrenset til denne prosessen og utsatt for endring ved tilgang på mer informasjon. Vår beste løsning fra denne undersøkelsen kan derfor variere med andre datasett.

Videre forskning kan være å undersøke benyttede algoritmer på andre datasett fra en bioprosess. Alt i alt, gir denne studien en innsikt i hvordan maskinlæring, spesielt semi-veiledet maskinlæring, kan predikere kvalitetsmålinger basert på data fra biokjemisk prosess-industri.

Abstract

Machine learning (ML) can find creative solutions to make processes more effective in various fields. The biochemical process industry is a place where continuous data is collected. The datasets from these industries are quite large, but they won't always be available as a complete dataset. One of the biggest problems in the industry is to get a fully labeled dataset. However, it is quite time-consuming and expensive, because the labels are collected by taking some samples to a laboratory.

However, applying machine learning algorithms to optimize processes requires a dataset with a certain criterion. To run supervised algorithms, the models require labeled data, which is not easily accessible. Thus, this study explores other ways of taking advantage of datasets that do not fulfill the criteria. Semi-supervised learning is a machine learning technique that can be applied in incomplete datasets. This has the potential to learn on both labeled and unlabeled data, which reduces the requirement of labeled data.

Furthermore, this research also aims to integrate data from a spectroscopic sensor to measure any improvements in the algorithm's predictions. For this research three SSL algorithms were implemented. These were compared to 3 supervised algorithms to identify the prediction quality.

This research is applying real-world dataset from Bioco AS. This is a biorefinery that works with processing raw materials from turkey and chicken. The dataset contains information from the process from several weeks. Every week there were tested different kinds of enzymes on the raw material, to understand how final product was. There were two different kinds of productions in a week, when the data was collected. The first one production was about testing different kinds of enzymes to the material, while the second production was to create the standard production that they have always made. When the labeled data was collected from the laboratory, they mainly focused on getting measurements mainly from the first production. This was to understand the chemical structure in the material. However, this made a bias in the labeled data in dataset. The labels in the dataset were not evenly distributed. The spectroscopy data was collected by a NIR sensor, which was temporarily placed at Bioco to measure more detailed data about the raw materials.

To measure the quality of the algorithm evaluation metrics were used. Since this is a regression problem, the following three metrics were used to evaluate the model. They were RMSE, R^2, MAE and MAPE. the thesis was based on a framework called CRISP-DM which is used to methodically break down the task.

The results from the research indicated the supervised algorithms performed better than the semi-supervised algorithms, even when it is limited labels. After comparing the effects of NIR-sensor data, the research further shows that the algorithms with data NIR outclassed the same algorithms without the spectroscopy data and only relying on the traditional process sensors.

A MCDA analysis was used to identify which methods that could be implemented in the industry today. After getting the results we had to test it to find out which model worked the best. Because the details of the company products are confidential, the following conclusions are limited to this research. The findings show that, even if SSL have theoretical promises, supervised algorithms outperform semi-supervised algorithms.

Further research is to implement the algorithms from this research in different dataset from the biochemical process industry. The research also advice to integrate spectroscopy data when using machine learning to predict on data from the process industry.

Blank Side

Innhold

Blank Side

Figurer

Tabeller

Blank Side

Ordliste

Begrep (Norsk):	**Terminology (English):**
Arbeidsminne	Random-Access Memory (RAM)
Beslutningsgrense	Decision Boundary
COREG	Regression with Co-Training
Deler	Folds
Etikettpropagering	Label Propagation
Fiolinplott	Violinplot
Forklaringsvariabler	Features
Forsterket læring	Reinforcement learning
Forklaringsvariabler	Features
Gjentagende Stratifisert K-Fold	Repeated Stratified K-Fold
Gjenværende punkt	Residual point
Ikke-veiledet læring	Unsupervised Learning
Informasjonsutbytte	Information Gain
Klyngeanalyse	Clustering
Knapphet heller Sjeldenhet	Scarce
Ladningsplott	Loading Plot
MAE	Mean Absolute Error
Manglede data	Missing data
MAPE	Mean Absolute Percentage Error
Marginfeil	Margin of Error
Markert	Labeled
Markert data eller Merket data	Labeled Data or Targets
MCDA	Multi-Criteria Decision Analysis
Med-Læring	Co-Training
Metrikk	Metric
MSE	Mean Squared Error
Multikollineraritet	Multicollinearity

Ordliste fortsettelse

Begrep (Norsk):	**Terminology (English):**
Prinsipale komponenter	Principal Components
Programgrensesnitt	Application Programming Interface - API
Prinsipalkomponentanalyse	Principal Component Analysis
Pseudo-Markering	Pseudo-Labeling
RMSE	Root Mean Squared Error
Rutenettsøk	Grid Search
Selv-Læring	Self-Training
Semi-veiledet læring	Semi-supervised Learning
Søyle intervaller	Bins
SSR	Sum Squared Regression
SST	Total Sum of Squares
Støtte vektor	Support Vectors
Støttevektor Regresjonsmodell - SVR	Support Vector Regressor - SVR
Stratifisering	Stratify
Stratifisert K-Fold	Stratified K-Fold
Teppediagram	Rugplot
Terskelverdi	Threshold Value
Tilfeldig Støy	Random Noise
Tilfeldig Søk	Random Search
Tilpasning av forklaringsvariabler	Feature Engineering
Treffsikkerhet	Accuracy
Umarkert	Unlabeled
Urenhetsmål	Impurity Measure
Utliggere	Extreme Value or Outlier
Utvalg	Subset
Variabel Viktighet	Feature Importance
Veiledet læring	Supervised Learning

Blank Side

Kapittel 1

Introduksjon

1.1 Bakgrunn

Den biokjemiske prosessindustrien har en sentral rolle i produksjon av varer rettet mot konsumering for både mennesker og dyr. Produktene spenner fra livsviktige nødvendigheter som farmasøytiske midler til spiselige matvarer som yoghurt og dyrefôr. Dermed følger det et stort ansvar om at innholdet er kvalitetssikret [1, 2]. Kvaliteten vurderes blant annet for å forsikre at produktene oppfyller målgruppens krav og er kritisk for lønnsomheten til produsenten [3].

Et av de største problemene i industrien er at innsamlingen av målinger på kvalitet er både kostbart og tidkrevende [4]. Siden noen av målingene krever å analysere det kjemiske innholdet, er det behov for å utføre analyser eksternt på et laboratorium. En konsekvens av disse laboratoriemålingene, er at direkte målinger på kvalitet ikke er tilgjengelig før en god stund etter at produktene er ferdigprodusert. Sen kvalitetsvurdering vil føre til utsettelse av umiddelbar beslutningsevne, og håndtering av produktene både underveis og etter produksjonen [5, 6]. Siden innsamling av disse målingene er kostnadskrevende, er det for øvrig høyere terskel for å samle det i større mengder.

I produksjonen er nøyaktig overvåkning og kontroll av prosessen essensielt for å forsikre effektivitet og produktkvalitet. Prosessen blir kontinuerlig overvåket underveis ved hjelp av ulike måleinstrumenter, og blir registrert i form av data [7]. Typiske og tradisjonelle målinger kan være temperatur, trykk og vibrasjon, og refereres som sensormålinger. Hensikten med innsamlingen av data er vedlikehold og forsikring om at nødvendige prosesser foregår som forventet, i tillegg til å oppdage eventuelle problemer. Imidlertid beskriver de tradisjonelle overnevnte sensormålingene omgivelser rundt materialet, og ikke nødvendigvis det kjemiske innholdet direkte. Målingene kan dermed gi indikasjoner på at produktene oppnår spesifiserte standarder og forventinger, men gi mindre presise indikasjoner på selve produktkvaliteten.

En spektroskopisk sensor kan analysere den kjemiske sammensetningen i materialet, uten å avbryte den kontinuerlige prosessen [8]. Informasjonsutbyttet i denne målingen gir innsikt i materialets innhold i prosessen. Målingene gir mer informative data knyttet til kvaliteten på råmaterialet, sammenliknet med de typiske temperatur- og trykkmålingene. Av den grunn kan de betraktes som direkte målinger på produktkvalitet. Inkludering

av en spektroskopisk sensor vil derfor kunne forbedre kvaliteten på informasjonen om produktkvalitet samlet av sensormålingene i prosessen. Nær infrarød måler (NIR) er en av flere sensorer som faller innen denne kategorien for måling av spektroskopisk data [9]. Integrering av en slik sensor som et steg i produksjonen, kan bidra til å øyeblikkelig tilgjengeliggjøre vesentlig informasjon som er knyttet til produktkvalitet tidligere i prosessen.

"Soft-sensor" er en modell designet til å predikere vanskelige målinger som produktkvalitet ved å ta i bruk historiske data av lettere anskaffede målinger. Sammenliknet med laboratoriemålinger blir sensormålinger ansett som lett tilgjengelige målinger, fordi de kontinuerlig monitoreres. Modellen baserer seg på å skape relasjoner mellom sensormålingene og laboratoriemålingene samlet i et datasett, ved bruk av maskinlæring [10, 11]. Bruk av målinger som har blitt samlet underveis, gjør det mulig for en soft-sensor å vurdere kvaliteten på produksjonen uten å flytte sluttproduktet til en ekstern enhet [12]. Hensikten med å bruke en slik modell er å minimere tidkrevende og kostbare laboratoriemålinger og heller gi en indikasjon på hvordan kvaliteten på råmaterialet kan være. Ved å forstå relasjonene mellom målingene, kan modellen også gi mer innsikt i hvordan styringsparametere påvirker produktkvalitet.

Tradisjonelt sett er soft-sensorer basert på modeller av veiledede maskinlæringslæringsmetoder [13]. Imidlertid er modellene innen veiledet læring avhengig av store mengder data og et fullstendig markert datasett til å predikere med god nøyaktighet. Et fullstendig datasett krever at alle sensormålingene er knyttet til minst én laboratoriemåling for å ha et referansepunkt. Data uten referansepunkt blir referert som umarkerte data. Sensormålingene måles kontinuerlig og er enkelt å samle inn. Imidlertid er terskelen for innsamling av laboratoriemålingene vesentlig høy. Dermed vil tilgjengeligheten av et fullstendig datasett være betydelig minimal [14].Som en konsekvens benyttes kun markerte data, som er en begrenset andel av all tilgjengelig data. Det fører til at store deler av innsamlede data, som er umarkerte, forblir ubenyttet.

Semi-veiledet læring er en alternativ metode som tar nytte av et ufullstendig datasett. I tillegg til å bruke markerte data, kan den ta nytte av informasjonen som er i umarkerte data til prediksjon. Til tross for dette er det fortsatt begrensninger rundt tilgjengeligheten av algoritmer innen slik type læring som kan predikere kontinuerlige verdier. Ettersom kvalitetsmålingene er kontinuerlige, er soft-sensor av semi-veiledede regresjonsmodeller lite utbredt. [15, 16].

1.2 Formål og Motivasjon

Formålet med oppgaven er å evaluere prediksjonsytelsen til ulike typer algoritmer som kan muliggjøre anvending av maskinlæring (ML) på data fra biokjemisk prosessindustri. ML gjør det mulig å predikere kvaliteten på bearbeidede produkter. Hvis potensialet i anvendelsen blir realisert, kan det frembringe nye indikasjoner på informasjon om produktkvalitet i en tidligere fase av produksjonen. Dette kan brukes til beslutningsstøtte og styring basert på sanntidsinformasjon.

Studien skal fokusere på evaluering av hvordan veiledede og semi-veiledede algoritmer presterer på industrielle og biokjemisk data. På grunn av varierende tilgang på sensormålinger og laboratoriemålinger i industrien, er det ønskelig å finne algoritmer som utnytter mer av all tilgjengelig data. Det er basert på at bruk av mer data potensielt kan øke mulighetene for forbedret kvalitet på prediksjoner. Veiledede algoritmer er avhengige av at alt innhold i datasettet som benyttes har referansepunkter. Det samles inn store mengder data kontinuerlig av sensormålinger uten referanser i industrien som vil forbli ubenyttet av veiledede algoritmer.

Det vil derfor være interessant å undersøke hvordan semi-veiledede algoritmer, som kan bruke data uten referanser, presterer på industrielle data. Med tilgang på mer data uten krav om referanser, kan det også bidra til å redusere kravet om innsamling av laboratoriemålinger til modellering. Dette kan gjøre metoden til en kostnadseffektiv alternativ. Oppgaven har dermed et mål om å kartlegge hvordan algoritmer av de ulike læringstypene vil prestere under forskjellige omstendigheter, som innebærer å eksponere dem for ulike mengder av data.

Et sekundært formål er å undersøke hvordan inkludering av spektroskopisk data kan bidra til å forbedre prediksjoner av kvalitetsmålinger. Siden spektroskopisk data inneholder informasjon knyttet til kvalitetsmålinger i industrien, vil det være interessant å undersøke hvordan inkluderingen av slik data vil påvirke de ulike algoritmene. Imidlertid kan tilgang på slik data kreve investering i denne type sensor dersom det ikke er en integrert del av prosessen. Dermed er det ønskelig å undersøke om forbedringen av algoritmene kan veie opp for eventuelle investeringer i en sensor som måler spektroskopisk data.

Studien kan på lenger sikt bidra til å gi beslutningstakere i industrien bedre verktøy til å forbedre håndtering av produkter basert på prediktert kvalitet. Utvikling og integrering av semi-veiledet modell som benytter seg mer av de innsamlede data i prosessen, kan også bidra til å forstå og forbedre prosessen i ytterligere grad.

1.2.1 Problemstilling og forskningsspørmål

Studiet er utarbeidet med forsøk på å sette søkelys på hvordan semi-veiledet modell og investering i spektroskopisk sensor kan bidra til prosessoptimering ved å øke prediksjonspresisjon på produktkvalitet. Oppgaven undersøker om maskinlæringsalgoritmer av både veilede og semi-veiledede metoder, med og uten tilgang på spektroskopisk data, kan benyttes som verktøy for å forbedre vurderinger og beslutninger tatt på produktkvalitet i den biokjemiske industrien. Med utgangspunkt i de identifiserte problemene om ubenyttede data og lite informative sensormålinger i den biokjemiske industrien, samt utgangspunkt i beskrevet formål, er følgende problemstilling utarbeidet. Den er som følgende:

Hvordan kan regresjons-algoritmer basert på semi-veiledet læring, bidra til å forbedre prediksjon av produktkvalitet i biokjemisk prosessindustrielle data sammenlignet med veiledet læring, ved å utnytte umarkerte data og spektroskopisk måleteknologi?

Problemstillingen baserer seg på følgende påstander:

- Semi-veiledede algoritmer vil bidra til mer presis prediksjon av produktkvalitet i biokjemisk prosessindustri sammenlignet med veiledede algoritmer, med tilgang på større datagrunnlag.
- Spektroskopisk data om kjemisk innhold i materiale vil gi bedre prediksjon av kvaliteten på sluttproduktet.

1.2.2 Forskningsspørsmål

For å kunne besvare problemstillingen, er det avgjørende å bryte det ned til relevante forskningsspørsmål. Spørsmålene bidrar til en systematisk tilnærming og forsikring at forskningen utføres i en retning med klare og definerte mål. I lys av problemstillingen, er det dermed blitt utledet og gjort forsøk på å besvare følgende forskningsspørsmål:

Forskningsspørsmål 1: I hvilken grad skiller prediksjonsytelsen av semi-veiledede modeller seg fra klassiske, veiledede modeller?

Forskningsspørsmål 2: Hvilke muligheter og utfordringer er knyttet til informasjon i umarkerte data som påvirker ytelsen til semi-veiledede modeller?

Forskningsspørsmål 3: Hvordan endres prediksjonsnøyaktigheten på produktkvalitet ved å benytte data av spektroskopiske målinger?

Forskningsspørmål 4: I hvor stor grad kan prediksjonsnøyaktigheten til maskinlæringsalgoritmer med tilgang på spektroskopisk informasjon, bidra til økt verdiskapning?

Det siste forskningsspørsmålet fokuserer på den økonomiske vinklingen av å implementere en potensiell soft-sensor med tilgang til spektroskopisk data. Soft-sensor kan bidra til både forbedring av beslutninger basert på produktkvalitet og muligens samle informasjon som kan forbedre slve produktkvaliteten. Imidlertid kan algoritmer av både veiledede og semi-veiledede læringer kreve ulike mengder markerte data for å kunne predikere med god presisjon. Dermed kan det variere med kostnader forbundet med datainnsamling av

spesielt laboratoriemålinger som referanser til å markere data. Undersøkelse av forskningsspørsmålet skal vurdere økonomiske aspekter ved mulig integrering av en prediktiv modell og spektroskopisk sensor i prosessen. Deretter vil mulighetene for økt verdiskapning ved hjelp av prediksjoner på produktkvalitet vurderes, dersom det veier opp for tilknyttede kostnader.

1.3 Avgrensninger og utfordringer

Oppgaven har fokusert på bruk av maskinlæringsalgoritmer, spesielt av semi-veiledede læringstyper, i biokjemisk prosessindustri. For å kunne realisere en slik undersøkelse var det nødvendig å benytte et virkelig datasett fra en bioprosess. Selv om andre biokjemiske prosesser kan være utstyrt med måleinstrumenter for temperatur og trykk, og muligens sensor for spektroskopisk data, vil produksjonen skille seg fra hverandre. Råmateriale som behandles vil ha egenskaper ved seg som er unikt for den spesifikke prosessen og kan involvere behandlinger som er unikt for det råmateriale. De relevante målingene på produktkvalitet vil også være forskjellige basert hva som produseres. Dermed er det kjent at analysene og anbefalingene nødvendigvis ikke er generaliserbare til stor grad.

Det har blitt utført en forenkelt MCDA-analyse i oppgaven (Se Seksjon 2.4). Analysen er først og fremst basert på begrenset informasjon om prosessen og de økonomiske akseptene ved den. I tillegg er analysen basert på vurderinger av kriterier, som kan involvere subjektive meninger.

1.3.1 Begrensing av arbeidsminne (RAM)

Grunnet begrensning av arbeidsminne (*eng. Random-access memory*) på egne PCer, var visse algoritmer begrenset til å benytte seg av en begrenset andel av datasettet som ble benyttet. En slik begrensning foresaket at modellene ikke ble utsatt for å utforske datasettets innhold fullstendig.

BHD begrenset til å ta akseptere enn begrenset andel av umarkerte data. Dermed forble modellens ytelse på fullstendig data ikke fullstendig utforsket.

1.4 Oppbygning av oppgaven

Denne oppgaven tar for seg en forskning om bruken av maskinlæringsalgoritmer, spesielt semi-veiledede regresjons-algoritmer på biokjemisk- og prosessindustrielt data. Oppgaven er videre strukturert for å dekke flere elementer. Kapittel 2, dekker relevante teorier innen maskinlæring og metodikker. Kapittel 3, dekker informasjon og beskrivelser om det aktuelle biokjemiske datasettet ble byttet til studiets formål. Kapittel 4, gjør rede for de metodiske valgene som ble utført for forskningens formål. Kapittel 5, presenterer relevante resultater fra forskningen, som videre diskuteres i Kapittel 6. Avslutningvis vil oppgaven avrundes med Kapittel 7, som er konklusjon på hvordan forskningen som har blitt utført kan svare på de definerte forskningsspørsmålene og problemstillingen. I tillegg blir det vist til Vedlegg A, for nødvendige illustrasjoner og informasjon som støtter våre

valg og funn.

Kapittel 2

Teori

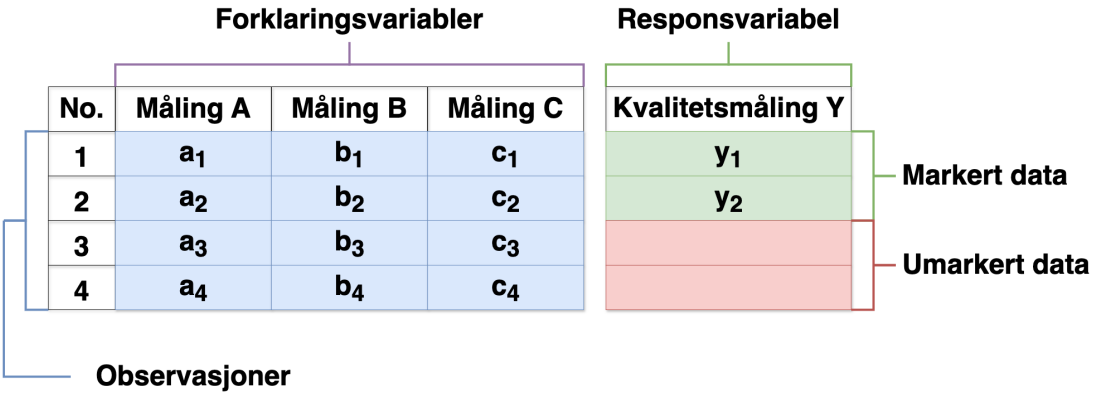
2.1 Grunnleggende maskinlæring

Maskinlæring er en gren innenfor kunstig intelligens og fundamentalt innenfor datavitenskapen. Det omhandler trening av datamaskiner ved hjelp av algoritmer og statistiske metoder til å utføre gitte oppgaver og avdekke verdifull innsikt i et datasett. Målet er å etterligne funksjonen til en menneskehjerne, ved å gradvis forbedre prestasjonsevnen basert på tilbakemeldinger fra tidligere data [17]. Opplæringen er basert på strukturen og eventuelle parametere til en algoritme. Resultatet er en prediktiv modell basert på gitt data. Det er dermed et krav på et datasett som modellen kan bygge et grunnlag på [18].

2.1.1 Utforming av datasett

Informasjonen og strukturen i et datasett kan inndeles i komponenter som observasjoner og variabler. De horisontale radene i datasettet består av observasjoner og representerer individuelle tilfeller eller enheter. Det kan gjelde en person, et tidspunkt eller en instans av et eksperiment. De vertikale kolonnene består av variabler og representerer ulik informasjon eller egenskaper hos observasjonene [19].

Variablene kan inndeles i to typer: forklarings- og responsvariabler. Informasjonen i responsen representerer et utfall eller resultat. Formålet med en maskinlæringsmodell er å kunne predikere denne typen av informasjon. Modellen baserer seg på en algoritme og bygger forståelse for å se sammenhenger mellom forklaringsvariabler og responsvariabler. Innholdet i forklaringsvariabler benyttes for å forklare informasjonen representert i responsen. Eksempelvis er konsentrasjonen av vann i et sluttprodukt et resultat av interesse. Informasjon om størrelse og vekt på produktet kan brukes til å forklare vannmengden i den. Forklaringsvariabler blir omtalt som prediktorer.



Figur 2.1: Denne figuren illustrerer strukturen til et datasett. Her blir det vist 3 forklaringsvariabler og 1 responsvariabel Y. Totalt 4 observasjoner er presentert. De øverste to radene med data, fremhevet i grønt er markerte data. De nederste to radene med data, fremhevet i rødt er umarkerte data.

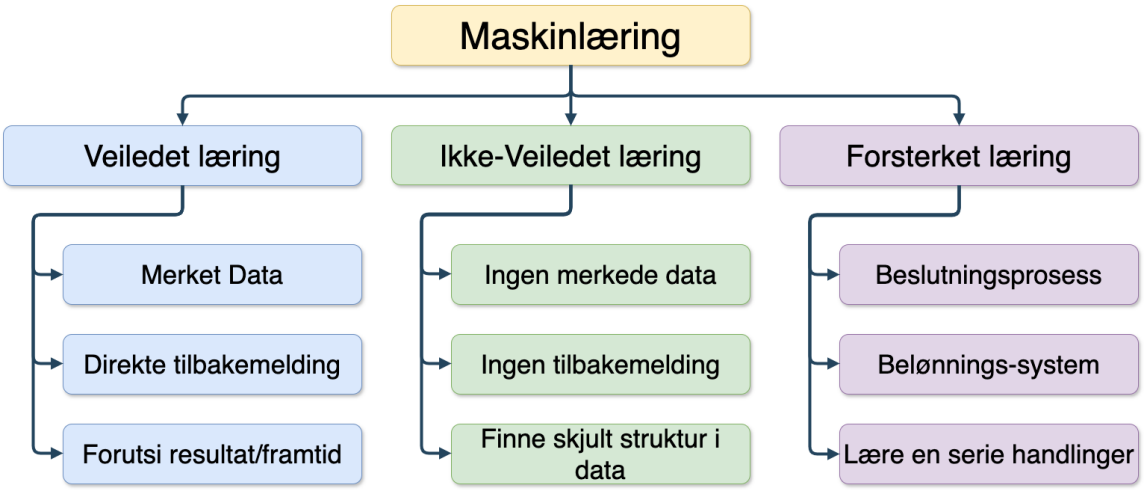
Innholdet i datasettet kan skilles mellom markert (*eng. Labeled*) og umarkert (*eng. Unlabeled*) basert på tilgjengeligheten av informasjonen representert i responsvariabelen. En observasjon er tildelt en etikett dersom informasjonen den beskriver er representert og tilgjengelig i responsen. Figur 2.1, viser et eksempel på datasett. Observasjon 1 og 2 har etiketter i responsvariabelen og er ansett som markerte observasjoner. Imidlertid mangler observasjon 3 og 4 etiketter, og betraktes som umarkerte observasjoner. Informasjonen de skal representerer er ukjent og ikke tilgjengelig i responsen. Ved fravær av informasjon i responsvariabler, omtales datasettet som umarkerte data. Det kan også gjelde ved mangel på en responsvariabel. Mangel eller utilgjengelig informasjon i et datasett kan representeres på mange former som blant annet:

- NaN - (Not a Number)
- () - Tom streng
- None - Ikke-eksisterende verdi

Informasjonen i variablene kan enten være kvalitativ eller kvantitativ [20]. Kvalitativ data er numerisk informasjon som temperatur og vekt. Slik type informasjon følger en naturlig hierarkisk orden. Det er gitt at temperatur på 30 grader celsius er *varmere* enn temperatur på 30 grader celsius. Kvalitativ data er kategorisk informasjon som kjønn og type råstoff. I motsetning til kvalitativ data er hierarkisk ordre avhengig av type kategorier. Det skilles hovedsakelig mellom ordinal og nominell data. Data av ordinal type kan rangeres i forhold til hverandre og følger en naturlig rekkefølge. En bukse med størrelse M er ansett som *mindre* enn en størrelse L
. Nominell data derimot har ingen intuitiv og naturlig måte å rangere informasjonen på. Eksempelvis kan ikke mann og kvinne rangeres på samme intuitive måte som buksestørrelser.

2.1.2 Ulike typer maskinlæring

Modellens evne til å lære på gitt data og utføre komplekse oppgaver, er avhengig av hvilken type metode den har blitt opplært etter. Generelt sett finnes det tre former for læring innen maskinlæring: veiledet læring, ikke-veiledet læring og forsterket læring [17]. I denne oppgaven vil det i hovedsak legges vekt på veiledet og ikke-veiledet læring. Figur 2.2 illustrerer de ulike læringsmetodene:



Figur 2.2: Denne figuren illustrerer tre retninger innen maskinlæring og viser til typiske trekk ved læringene [17].

Veiledet læring

Veiledet maskinlæring (*eng. Supervised Learning*) er en av flere metoder innen maskinlæring, som vist i Figur 2.2. Det innebærer utvikling av modell på et fullstendig datasett som kun består av markerte observasjoner. Metodikken forutsetter dermed at all informasjon i responsen er kjent og tilgjengelig. Opplæringen foregår dermed på definerte mål for endelig modell. Under opplæringen vil modellen lære å finne sammenhenger mellom informasjonen i prediktorene og responsen. Modellen justeres og tilpasses for å minimere feil mellom predikert og faktisk informasjon i responsen. For at modellen skal kunne predikere optimalt er det nødvendig å trene godt nok i treningsfasen og deretter lage en generaliserende modell som fungerer på varierende data, før den kan bli testet på et testsett [21].

Problemtyper innen veiledet maskinlæring kan inndeles i to hovedkategorier: klassifisering og regresjon. Forskjellen mellom problemene ligger i hva slags type informasjon som er representert i responsen. Dersom den består distinkte klasser eller kategorier, vil det defineres som klassifiseringsproblem [22]. Formålet med en modell vil være tildeling av en kategori, basert på informasjonen i forklaringsvariablene. Det kan være binært problem, som skille mellom "godkjent kvalitet" og "ikke godkjent kvalitet", eller multiklasseproblem som skille mellom "kylling", "hane" eller 'høne'. Dersom informasjon i responsen er kvantitativ og kontinuerlig, er det et regresjonsproblem [22]. I slike tilfeller er målet å utvikle en modell som kan predikere numerisk informasjon. Eksempler på regresjonsproblem kan være estimering av vekt til en person eller innholdet av mengde proteiner i et produkt.

Ikke-veiledet læring

Ikke-veiledet læring (*eng. Unsupervised Learning*) omhandler å trene en modell uten at definerte svar er tilgjengelige under opplæringen. Modellen må være i stand til å oppdage strukturer i datasettet uten veiledning [17]. Uten informasjon i responsvariabelen, kan det forsøkes å avdekke mulige mønstre i data som skiller ulike grupperinger fra hverandre. Det kalles klyngeanalyse (*eng. Clustering*), der formålet er å identifisere og samle objekter eller variabler ut fra graden av likhet. Data som tolkes lignende av modellen vil samles og ulik data vil skilles fra hverandre [23].

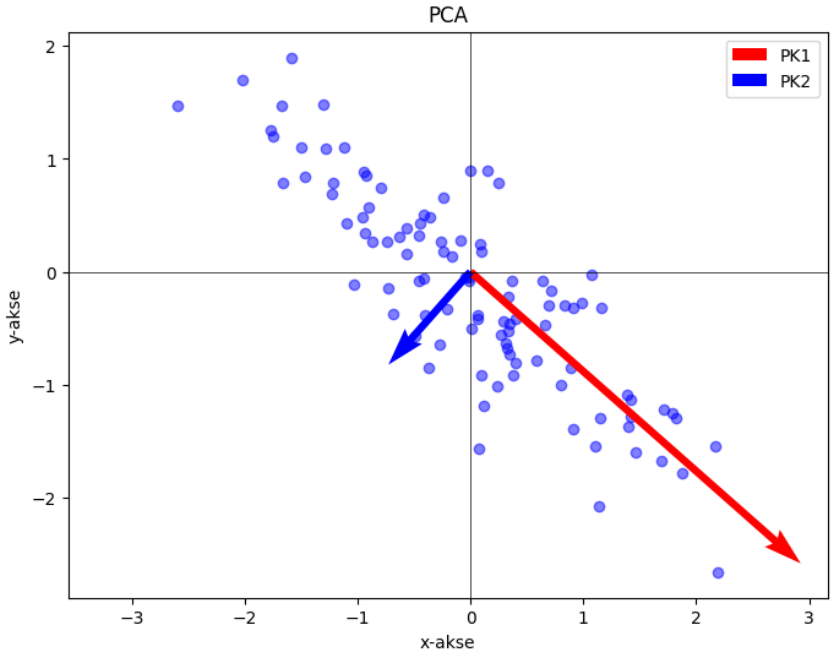
PCA - Prinsipalkomponentanalyse

PCA, kjent som prinsipalkomponentanalyse (*eng. Principal Component Analysis*), er en teknikk som blir brukt innen ikke-veiledet maskinlæring til å forenkle data med høy dimensjonalitet. Denne metoden finner mønstre i et datasett ved å analysere korrelasjonen mellom variablene [24]. PCA transformerer dimensjonen og forklarer variasjonen i datasettet med nye variabler.

I et datasettet er det normalt at flere av variablene korrelerer med hverandre. Dette fører til at de forklarer lik informasjon eller variasjon. Når PCA lager nye variabler bevarer de mye av informasjonen i data og beholder høy varians til tross for at dimensjonen kan minke. De nye variablene er vektorer og kalles prinsipale komponenter (*eng. Principle Components*).

Figur 2.3 viser et eksempel som illustrerer vektorene, $PK1$ og $PK2$, i to dimensjoner. Hver komponent tar en vektet kombinasjon av alle variablene fra det opprinnelige datasettet. Hver vekt forklarer variabelens innflytelse på hele komponenten. Variablene blir vektet på en måte som fører til at komponentene ikke korrelerer med hverandre. Ved å ta hensyn til at komponentene ikke korrelerer, sørger man for at de ikke forklarer lik informasjon. Dermed vil hver PK-vektor fremheve variasjonen tydeligere [25]. I det transformerte datasettet vil $PK1$ ha størst varians. Deretter vil variansen minke for hver ny vektor som opprettes. Siden de nye vektorene tar i bruk store deler av informasjonen i det opprinnelige datasettet, minker det behovet for å ha like mange vektorer i det transformerte datasettet. Dermed kan dimensjonen til det transformerte datasettet defineres slik $k \leq d$. Verdien k representerer antall vektorer i det transformerte datasettet, mens verdien d er antall vektorer i det opprinnelige datasettet.

De ulike vektingene for hver av variabelene fra PCA kan visualiseres ved hjelp av en ladningsplott (*eng. Loading Plot*). Når visualiseringene til alle komponentene sammenliknes, gir det muligheten for å forstå hvilke variabler som har en stor innflytelse i det opprinnelige datasettet. Dette hjelper med å øke dataforståelsen. Et eksempel på en ladningsplott er lagt i vedlegget A.7, ved bruk av datasettet til oppgaven.



Figur 2.3: Figuren over ble visualisert i Python ved hjelp av biblioteket Scikit-learn (se Vedlegg A.3.1). Denne figuren viser en illustrasjon av hvordan de nye ortogonale aksene "PK1" (rød) og "PK2" (blå) i en PCA-analyse gjennomføres i et datasett.

Semi-veiledet læring

Veiledet læring og ikke-veiledet læring representerer to ulike tilnærminger for å behandle og predikere datasett. Likevel er det enkelte tilfeller der datasettet kan inneholde både merkede og ikke-merkede datapunkter. Da fremstår semi-veiledet læring (*eng. Semi-supervised Learning*)som et fordelaktig alternativ for modellopplæring. Semi-veiledet læring er en teknikk som kombinerer metoder fra begge tilnærmingene der få markerte data i et datasett benyttes med varierende mengder av umarkerte data [26]. Innen semi-veiledet læring finnes det ulike teorier og metodikker som lærer og finner mønstre på ulike måter.

Selv-læring og Pseudo-markering

Selv-læring (*eng. Self-training*) er en metode innen semi-veiledet læring, der modellen genererer egne markerte data i treningsfasen. Modellene starter med å trene først på et utvalg (*eng. Subset*) av de merkede datapunktene i datasettet. Deretter vil modellen analysere umerkede data for å kunne sette midlertidige markeringer på datasettet, kjent som pseudo-markeringer [27]. Innad i de ulike modellene i selv-læring er det forskjellige teknikker for hvordan de umerkede datapunktene blir tatt hensyn til.

Pseudo-markeringer (*eng. Pseudo-labeling*) blir brukt dersom det hjelper modellen til å prestere bedre over tid. I denne prosessen brukes en terskelverdi (*eng. Threshold Value*)for å kunne skille de umarkerte datapunktene mellom kvalifiserbare og ikke-kvalifiserbare. Terskelverdien settes ved bruk av en feilmetrikk eller et konfidensintervall [28]. Modellen trenes deretter på en kombinasjon av både de pseudo-markerte datapunktene og de opprinnelige merkede datapunktene med mål om å styrke modellens robusthet [29].

Prosessen er iterativ, der et kvalifisert utvalg av pseudo-merkede data blir en del av det markerte treningssettet etter hver runde. Siden det markerte treningssettet kan endres etter hver runde, vil det også settes nye pseudo-markeringer på de umarkerte datapunktene. Denne iterative prosessen forsetter fram til prediksjonsytelsen til modellen ikke forbedrer seg eller mangel på umarkerte datapunkter overgår satt terskel.

Med-læring

Med-læring (*eng. Co-training*) har like grunnprinsipper som selv-læring, men skiller seg ut på visse områder. Hovedsakelig er det to modeller som jobber parallelt for å løse et problem, motsetning til selv-læring som jobber selvstendig. Modellene som baserer seg på med-læring, får ulikt syn på datapunktene og pseudo-markerer de umarkerte datapunktene på sin måte. Begge modellene legger vekt på tilliten til pseudo-markeringen som er gitt til datapunktene hver for seg [30]. Vektleggingen avgjør den endelige pseudo markeringen som blir gitt til de umarkerte datapunktene i datasettet. Etter at begge modellene tar en vurdering hver for seg, vil de deretter sammenlikne tilliten til pseudo-markeringen [27]. Slik jobber de sammen parallelt og konkluderer seg fram til en endelig pseudo-markering. Denne iterative prosessen fortsetter fram til treningssettet blir fullstendig pseudo-markert. Dersom tilliten til pseudo-markeringen ikke når en viss beslutning eller ikke tilfredsstiller et forutbestemt krav, kan prosessen avsluttes tidligere.

Grafbasert etikettpropagering

Etikettpropagering (*eng. Label Propagation*) er en annen teknikk innen semi-veiledet læring som brukes for å etikere umerkede data ved hjelp av merkede data. Her vil hvert enkelt datapunkt bli sett på som en node. Denne teknikken antar at de nærmeste datapunktene har like etiketter. Dermed gis det etiketter til de umarkerte datapunktene ut ifra etikettene til de nærmeste markerte datapunktene [31]. Når man skal finne avstanden mellom de ulike datapunktene på grafen, blir det benyttet ulike avstandsformler [32]. I dette forsøket ble blant annet euklidsk avstand brukt, som er en annen variant av minkowski formlen (se Formel 2.6).

2.1.3 Trening, kryssvalidering og testing

Trening og testing av data

For gunstig opplæring deles det benyttede datasettet i to deler: treningssett for opplæring og et testsett for å evaluere modellens prestasjon på ukjent data [33].

Når maskinlæringsalgoritmer skal modelleres, vil datasettet ofte bli oppdelt i to separate deler. Her kalles den første andelen for treningsdata og den andre delen kalles testdata. Oppdelingen blir gjennomført slik at algoritmen først skal kunne trene seg opp ved å lære seg mønstre og sammenhenger på avsatt på en viss mengde data [34]. Deretter skal den opptrente modellen teste evnen til å predikere på nye og usette data. Testsettet skal være uavhengig av treningsfasen, for å muliggjøre en objektiv evaluering av modellen og gi en realistisk indikasjon på ytelse på ukjent data [35].

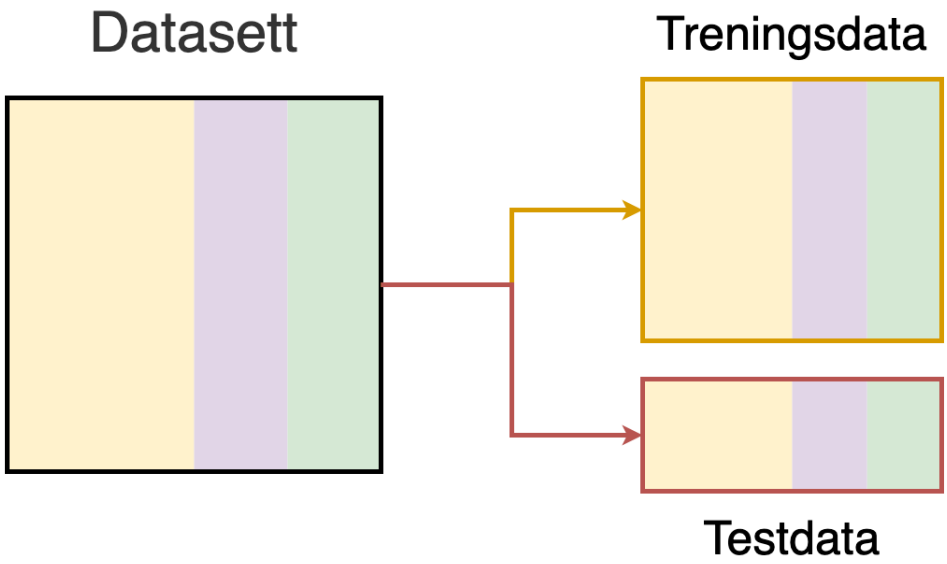
Undersøkelse på testdata vil virke som en simulering på hvordan modellen hadde prestert på fremtidig data som ikke hadde vært tilgjengelig i øyeblikket. Observasjoner gjort under undersøkelsen kan benyttes til å identifisere mulige feil og svakheter i modellen. Justering av forutsetninger og parametere til modellene blir basert på vurderinger av undersøkelsen. Dette bidrar til å forbedre modellens evne til å predikere ytterligere [17]. Dersom testsett benyttes til slikt formål, vil den lenger ikke være uavhengig. Justeringer basert på informasjon fra testdata, kan gi videre evalueringer et falskt inntrykk av hvor godt modellen presterer.

Over- og undertilpasning

Overtilpasning og undertilpasning er to typiske utfordringer maskinlæringsmodeller utsettes for i treningsfasen. Hensikten til en modell er å oppnå god generaliseringsevne som kan predikere godt på ny og usett data. Når parameterne til modellene justeres, optimeres de for å forbedre ytelsen på prediksjonene. Når modellene trenes opp på et datasett og tilpasser seg for absolutt alle detaljene til datapunktene kan det føre til dårligere prediksjonsytelse på usett data. Når det er store avvik mellom prestasjonene i trening- og testfasen kan det tyde på modellen har overtilpasset seg på informasjonen i treningsdataen. Modellen er altfor kompleks og henger seg opp i støy og mindre viktige detaljer [36]. Dette kan gå på bekostning av modellens evne til å gjenkjenne generelle sammenhenger i usett data av lik natur. Overtilpasning kan medføre til høy varians. Når modellen mislykkes til å finne sammenhenger i stor grad, kan det derimot føre til svak prediksjonsevne både i trening- og testfasen. Modellen er for enkel og er ikke i stand til å fange opp en større andel av de viktigste sammenhengende i data. Slike tilfeller kalles undertilpasning, og kan medføre høy bias.

Stratifisering

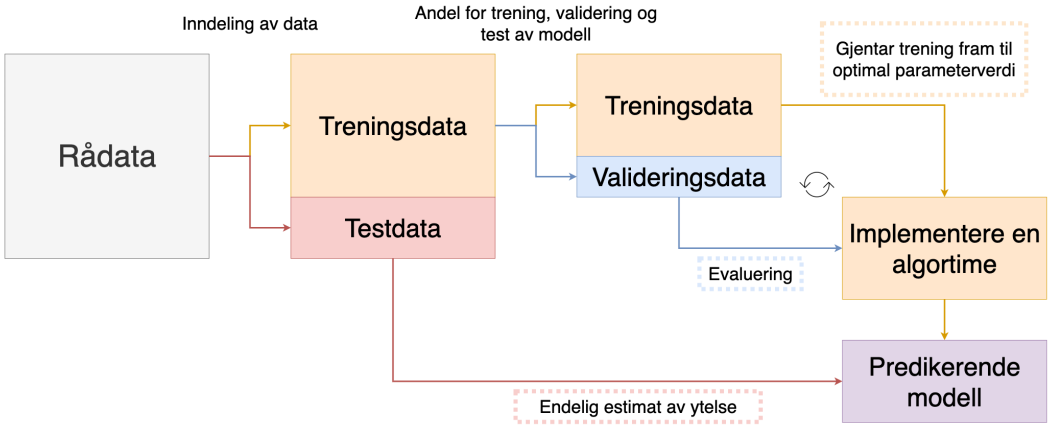
Informasjonen i datasettet kan inndeles i undergrupper basert på egenskap eller kategori. Fordeling av trenings- og testdata ved tilfeldig utvalg, kan forårsake over- eller underrepresentasjon av bestemte undergrupper. For at informasjonen i datasettet proporsjonalt representert i både trenings- og testdata, kan stratifisering (*eng. Stratify*) benyttes. Teknikken vil sørge for at den opprinnelige fordelingen av kategoriene i en spesifikk variabel blir opprettholdt ved fordelingen av datasett. I treningsfasen vil det sørge for at modellen trener på informasjon som er mer representativ for all innsamlet data. I testfasen vil det forsikre at modellen evalueres etter evnen til prediksjonen, basert på av data der undergruppene er representert. Se Figur 2.4 for illustrasjon.



Figur 2.4: Denne figuren illustrerer et datasett som blir separert i to deler: treningsdata og testdata. Separeringen sørger for at data fra testandelen ikke blir inkludert når algoritmer trener på treningsdata. De ulike fargene representerer diverse undergrupper i datasettet. Med stratifisert fordeling er fordelingen av undergruppene opprettholdt i både treningsdata og testdata.

Kryssvalidering

Holdout-metoden er en metode for å evaluere hvor godt modellen er til å generalisere på usett data. Dette gjennomføres ved å utelate et segment av data til evaluering og resterende til trening. Som vist i Figur 2.5, er metoden brukt for å separere det opprinnelige datasettet til trening- og testdata. Det er igjen benyttet til å splitte treningsdata til treningssett og valideringssett. Modellen trener på treningssettet og evalueres på valideringssettet. Sistnevnte anses som et midlertidig testsett. Fordelingen mellom trenings- og valideringsdata varierer ut ifra behov og er ikke en fast andel. Siden data skiller seg fra avsatt testdata, kan modellen benytte det til å justere parametere basert på undersøkelser av valideringen. Testsettet blir ikke benyttet for at det ikke skal påvirke objektiv vurdering av prestasjonsevnen til modellen. Ulempen med metoden er at modellen har mindre data å trene på og at metoden er sensitiv til fordelingen av det opprinnelige treningssettet. Størrelsen på treningsdata har stor betydning for modellens evne til å forstå mønstre og sammenhenger [17].



Figur 2.5: Denne figuren viser en illustrasjon av hvordan holdout-metoden splitter treningsdata til et treningssett og et valideringssett. Videre forklarer den hvordan modellen anvendes ved å repetere med ulike parameterverdier og teste på valideringssettet. Basert på kriteriene satt for prediksjonsevnen, vil modellen med et sett av parameterverdier testes på testdatasettet som har vært adskilt fra resten av prosessen.

K-Fold

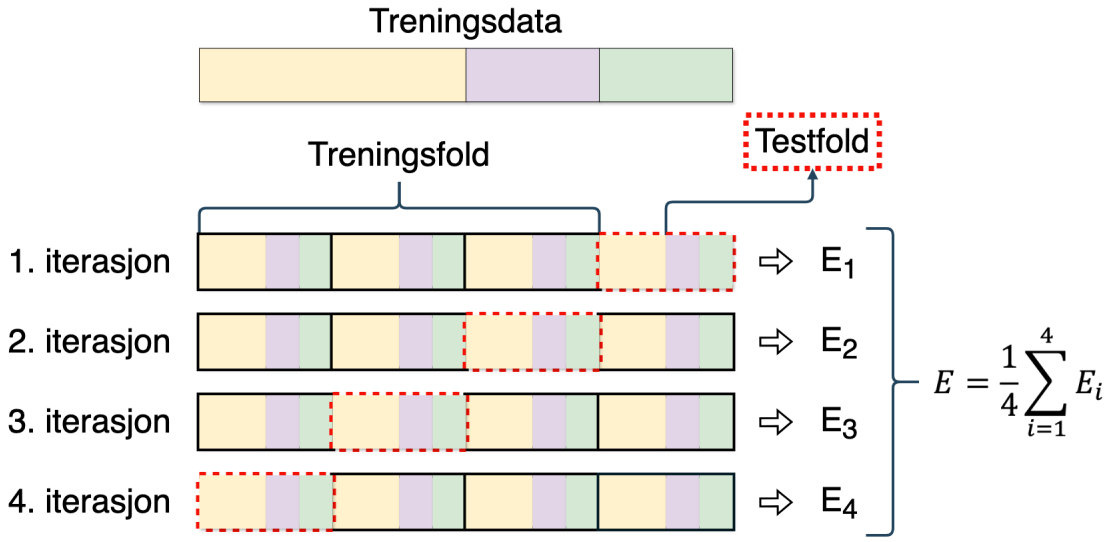
K-fold er en variant innen kryssvalidering, der treningssettet deles opp etter k ønskede antall deler (*eng. Folds*) av lik størrelse. Her vil 1 av k antall deler bli satt av til validering, mens de resterende $k-1$ andelene av treningssettet blir brukt til å modellere. Hvilken del av det opprinnelige treningssettet som blir valideringssett byttes på. Prosessen utføres iterativt k antall ganger gjennom treningsfasen [37]. Under den iterative prosessen vil hvert enkelt datapunkt i de opprinnelige treningssettet være en del av valideringssettet en eller annen gang mens modellen går igjennom iterasjonene. Den endelige vurdering vil være en gjennomsnittlige resultat basert på alle delresultater fra iterasjonene. Ut ifra tidligere ulike gjennomførte forsøk av både Kohavi R. og fra Marcot B. G. viser det seg at en k verdi på 10 gir en god balanse mellom bias og varians [38, 39].

Metoden kan ses på som en videreføring av "holdout"-metoden, der opprinnelig treningsdata blir oppdelt i flere enn to segmenter. På samme måte som "holdout"-metoden, holdes et av segmentene for validering utenfor trening av modellen. Metoden skiller seg ved at den midlertidige evalueringsprosessen er iterativ og gjennomføres til alle segmenter er benyttet til validering. Under hver iterasjon vil modellen trene på de resterende segmentene. Prosessen fortsetter til alle datapunkter i det opprinnelige treningssettet har blitt predikert.

K-fold kryssvalidering er et mer fordelaktig alternativ til "holdout-metode", da den er mindre sensitiv til en bestemt oppdeling. Siden alle datapunkter er involvert i alle dere av prosessen, er den ikke avhengig av oppdelingen. Ved å evaluere over flere segmenter, gir det også et mer robust estimat på modellens generaliseringsevne. Gjentagelse av trening- og validering i flere omganger, bidrar til å redusere variansen i evalueringen av ytelse. Dermed vil evalueringen være et mer pålitelig og robust estimat. Imidlertid er metoden mer beregningskrevende [17].

Stratifisert k-fold (*eng. Stratified k-fold*) er en variant av tradisjonell k-fold, der hensikten er inkludere stratifisering under oppdelingen av segmentene. Det fokuseres det på å bevare den opprinnelige fordelingen av undergrupper i en variabel i hvert oppdelte segment [40]. Uten spesifisering vil k-fold tradisjonelt sett fordele informasjonen i det opprinnelige data på en tilfeldig måte. Det kan føre til at informasjonsfordelingen og representasjonen av visse undergrupper er skjevfordelt i hvert segment. Stratifisert k-fold forsikrer at alle undergrupper representeres rettferdig ut ifra fordelingen i det opprinnelige treningssettet. På den måten vil ikke modellens evne til å oppdage viktige og generelle mønstre i data være avhengig av oppdelingen.

Gjentagende stratifisert k-fold (*eng. Repeated Stratified k-fold* er videreføring av stratifisert k-fold, der hensikten er å redusere avhengigheten av ytterligere oppdeling. Gjentagende stratifisert k-fold gjentar stratifisert k-fold n antall ganger på datasettet, med ny og tilfeldig stratifisert fordeling hver gang [41]. Metoden er spesielt hensiktsmessig når datasettet er begrenset. I slike tilfeller kan hvert segment inneholde en begrenset andel av tilgjengelige datapunkter. Med færre datapunkter å trene på, kan ytelsen variere mellom iterasjonene. En slik begrensing bidrar til større avhengighet til den tilfeldige fordelingen av datapunktene. I tillegg vil tradisjonell k-fold og stratifisert k-fold resultere i en bestemt oppdeling av data. For en enda mer robust og pålitelig vurdering av modellens ytelse, kan prosessen gjentas med ny fordeling etter iterasjonene avsluttes. Dermed blir hele prosessen til k-fold kryssvalidering gjentatt flere ganger, med nye fordelinger i segmentene for hver gjentakelse. Dette skaper både mer variasjon i treningsmateriale og lager flere unike valideringssett til modellen. Ved ha å lage flere folds basert på ulike kombinasjoner av treningsdataen, kan det minke uforutsigbarheten i data i tillegg til å redusere risikoen for overtilpasning [42].



Figuren viser kryssvalidering, i dette tilfellet 4-fold med stratifisering.

Figur 2.6: Denne illustrasjonen viser bruk av k-fold som teknikk for kryssvalidering. Her er det opprinnelige treningssettet oppdelt i 4 segmenter. Fordelingen av ulike undergruppene i opprinnelige treningsdata er bevart i hvert segment.Treningsdata har 3 klasser og representeres av distinkte farger i figuren. Forholdet mellom klassene i treningsdata (1. gul - 50 %, 2. lilla - 25 % og 3. grønn 25 %). Det samme forholdet er bevart i de ulike segmentene.

2.1.4 Metrikker for evaluering av regresjonsmodeller

Ytelsen til semi-veiledede regresjonsmodeller skal vurderes og sammenlignes med veiledede regresjonsmodeller. Når vi jobber med regresjonsmodeller, finnes det ulike metoder for å evaluere hvor godt modellen presterer. Dermed regnes det på evalueringsmetrikker som RMSE, R^2, MAE og MAPE for å få mer innsikt i robustheten til modellen. Disse er standard evalueringsmetrikker for regresjonsmodeller [43].

RMSE

RMSE (*eng. Root Mean Squared Error*) er en av de mest benyttede evalueringsmetrikkene for regresjonsproblemer. Det brukes som et mål for å evaluere kvaliteten på en modell, ved å beregne gjennomsnittlige avvik mellom opprinnelige data og predikert verdi [44]. Avstanden til dette avviket omtales som residual. RMSE vurderer det totale avviket med hensyn til alle residualene, med en generell formel:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y^{(i)} - \hat{y}^{(i)}\right)^2} \tag{2.1}$$

Først beregnes RMSE ved å sumerere kvadratet av residualene. Hensikten er å forsikre at negative og positive residualer ikke kansellerer hverandre. Ved kvadrering vil alle residualer være positive. En videre konsekvens av kvadrering er at betydeligheten av større avvik blir enda mer forsterket. Metrikken er dermed mer sensitiv til utliggere som ofte forårsaker større avvik [44]. Endelig score vil være basert på roten av samlingen av de kvadrerte avvikene.

Når RMSE verdien er nær null, viser det til lite avvik i modellen, dermed indikerer det at modellen er robust og har tilpasset data i god grad. En høy RMSE verdi betyr at modellen ikke representerer datapunktene godt nok og gir prediksjoner av lav kvalitet [45].

R^2 - Forklart varians

R^2 er en evalueringsmetrikk som gir et mål på hvor godt en regresjonsmodell kan forklare innholdet i data den prøver å predikere. Metrikken gir vurdering på graden av variasjonen i responsen som forklares av forklaringsvariablene i modellen. Det gis en score i et intervall. R^2-score har et intervall på $[0, 1]$ [46]. Formelen for R^2 er som følgende:

$$R^2 = 1 - \frac{\text{SSR}}{\text{SST}} \tag{2.2}$$

SSR (*eng. Sum Squared Regression*) er summen av kvadratet til avvikene mellom observerte og predikerte verdier. SST (*eng. Total Sum of Squares*) er totale summen av avvikene mellom observerte verdier og gjennomsnittsverdien i responsen. Formel 2.2 kan gjøres om til følgende:

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2} \tag{2.3}$$

Eksempelvis vil en høy R^2-score på 0,9 tyde på at 90 % av variasjonen i responsen er forklart av forklaringsvariablene til modellen. Høyere score er forbundet med god prediksjonsytelse. $R^2 = 1$ vil bety at modellen passer perfekt til data og kan forklare all variasjon i responsen. Imidlertid kan det gi en indikasjon overtilpasning og at modellen er for kompleks. Lave R^2-score kan indikere undertilpasning ved at lite variasjon kan forklares av modellen. Modellen kan være for enkel med dårligere ytelse [47].

Negativ R^2-score kan oppstå i de tilfellene der modellen har tolket data feil på en alvorlig grad. Det oppstår når SSR > SST. Når modellen passer dårligere til data enn hva en horisontal linje gjennom gjennomsnittet av responsen, kan det resultere i en negativ R^2-score. Modellen har dermed mislykket i å fange opp viktige variasjoner i responsen.

MAE og MAPE

MAE (*eng. Mean Absolute Error*) er en annen metrikk som brukes for å evaluere prediksjonsevnen til en regresjonsmodell. Det gjennomsnittlige avviket beregnes ved å summere absoluttverdien av avviket mellom observerte og predikerte verdier [48]. Følgende formel for MAE er:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{2.4}$$

MAE er mindre sensitiv til utliggere sammenliknet med RMSE (se Seksjon 2.1.4). Ved å ta absoluttverdi fremfor kvadrering, vil alle residualer gis lik vekting og betydning til den totale vurderingen. På den måten er det mindre sensitiv til utliggere [49].

MAPE (*eng. Mean Absoute Percentage Error*) er en alternativ måte å uttrykke det absolutte avviket på. Det gir prosentvis verdi på gjennomsnittlig avvik mellom predikerte og faktiske verdier. Metrikken regnes ved bruk av følgende formel:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{2.5}$$

MAPE beregnes ved å dele avvikene på det faktiske verdiene. På den måten vil det være et mål på relativ feil og kan dermed benyttes til å sammenligne modeller på tvers av datasett og uavhengig av responsen [50].

2.1.5 Optimering av hyperparametere

Basert på evalueringer av modellen, justeres den for at prediksjonsytelsen skal forbedres. Prosessen for målrettet justering kalles for hyperparameteroptimering. Hensikten med prosessen er å finne en god balanse mellom kompleksitet og generaliseringsevne. For å redusere problemer med tilpasninger i tillegg til å finne en balanse, benyttes diverse teknikker for hyperparameter-optimering. Optimeringen er forbundet med gitte evalueringsmetrikker som er definert på forhånd [51].

Algoritmer er utstyrt med et sett med parametere. Hyperparameteroptimering handler om å bruke ulike kombinasjoner av parametere for å finne den kombinasjonen som gir best utbytte [52]. Det finner ulike framgangsmåter for denne prosessen. Rutenettsøk (*eng. Grid Search*) og tilfeldig søk (*eng. Random Search*) er eksempler på to typiske teknikker innen optimering. Det har blitt vist at Bayesiansk optimeringsteknikk mer effektiv metode enn de to sistnevnte [53].

Kryssvalidering kan benyttes for å gi et bedre og mer robust grunnlag å basere optimeringen på. Teknikken k-fold kan bli brukt for å opprette flere versjoner fra det avsatte treningssettet. For hver ny fold som lages blir det satt at nye valideringssett, som parameterne kan bli testet på [17].

Dette er relevant når modellens parametere blir justert. Da har den behov for å teste den nye modellen flere ganger. Da er det viktig å ha flere datasett å jobbe med. Dermed vil k-fold få modellen til å unngå trening på samme del av treningsdatasettet flere ganger. Dette hindrer overtilpasning (se Seksjon 2.1.3) fordi man heller trener på ulike andeler av treningssettet hver gang. Deretter evaluerer modellen ved å observere kvaliteten på prediksjonsevnen til den nylig justerte modellen.

2.1.6 Algoritmer for forbehandling av data

LOCF - Last Obsersvation Carroid Forward

LOCF er en forkortelse som står for "Last Observation Carried Forward" [54]. Dette er en metode som implementeres når det er manglede verdier i datasettet. Denne teknikken blir ofte brukt ved tidsserieanalyser ved å analysere de siste verdiene før de manglende verdiene i datasettet. Manglede verdier i datasettet fører til dårligere kvalitet av data i de små utvalgene som blir plukket ut av datasettet. Dermed påvirker prediksjonsevnen til algoritmen [55].

LOF - Local Outlier Factor

Local Outlier Factor (LOF) er en metode for å oppdage avvik i datasett. Metoden fokuserer på identifikasjon av lokale ekstremverdier framfor globale, ved å analysere nærliggende naboer. Et punkt betraktes som ekstrem dersom det avviker fra sine lokal naboer. LOF tar hensyn til konteksten til datapunktet, og vurderer tettheten og nærheten til naboene. Datapunktet vurderes som ekstremverdi dersom tetthet og nærheten av naboer, avviker betydelige fra tetteheten og nærheten til naboenes naboer [56]. For å finne k nærmeste naboene, brukes distansemetrikker i Minkowski Formel 2.6.

CBLOF - Clusterbased Local Outlier Factor

CBLOF, også kjent som "Clustering-based Local Outlier Factor" er en teknikk innen maskinlæring som brukes for å oppdage ekstremverdier. Denne prosessen lager en klusteranalysemodell på et gitt datasett ved hjelp av kMeans algoritmen [57]. kMeans er en ikke-veiledet maskinlæringsalgoritme som går ut på å gruppere umarkerte data basert på deres likheter i k-antall grupper. Når CBLOF teknikken brukes, blir det markert flere klustre og de vil bli markert som "store" eller "små" klustre. Denne kategoriseringen brukes videre for å regne på terskelverdien som endelig avgjør om datapunktene er ekstremverdi eller ikke.

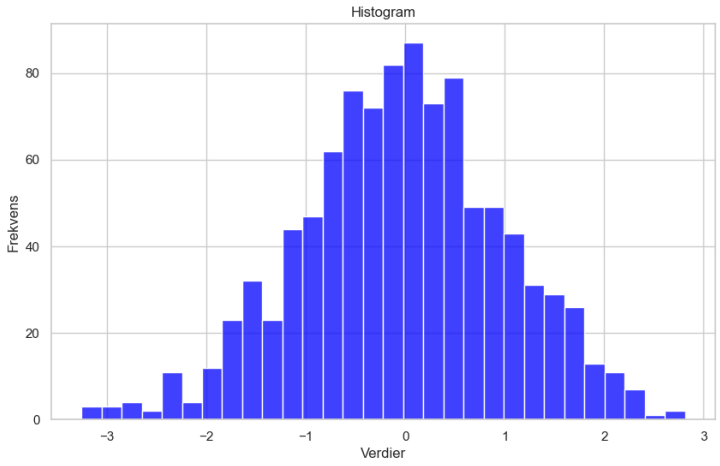
2.1.7 Visualisering av data

Visualisering spiller en sentral rolle i maskinlæring ved å fremstille informasjon for en helhetlig forståelse. Det er ikke bare et verktøy for å presentere resultater, men også en del forståelse av data, utvikling av modell og feilsøking. Mønstre og sammenhenger kan ofte forsvinne bak tall og verdier, dermed er visualiseringer er god måte å få en mer overordnet oversikt. Forskjellige teknikker har ulik hensikt og kan bidra til å avdekke informasjon om datasettet eller modellen. I denne seksjonen vises det til de ulike visualiseringsteknikkene som ble benyttet i oppgaven.

Histogram

Histogram brukes til å visualisere datapunkter i et to-dimensjonalt grafikkfelt. I grafen vil verdiene i de kontinuerlige variablene bli delt opp i ulike søyleintervaller (*eng. Bins*). På grafikkfeltet blir x-aksen delt etter størrelsen på intervallene. Y-aksen viser til frekvens eller prosentandel, avhengig av formålet med analysen. Størrelsen på intervallene

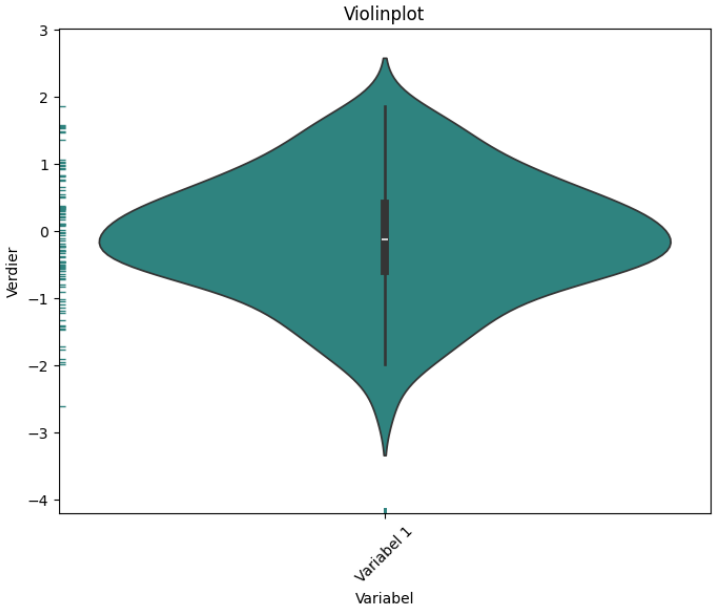
justeres ut ifra nøyaktigheten og mengden av data som er tilgjengelig. Deretter vil antall observasjoner av datapunktene bli telt opp og ført inn i tilhørende intervall. Intervallene forenkler kompleksiteten til å presentere data og fremmer andre mønstre i datapunktene. Mønstre som dette kan være krevende å gjenkjenne uten slik behandling. Histogrammet gir en visuell representasjon av fordelingen i data. Det kan gi informasjon om variasjonen i data ved se på spredning i fordelingen. Det gir også en formening om skjevhet i data ved å sammenligne hvordan de ulike søylene forholder seg til hverandre. Histogram kan dermed gi dermed innsikt i overordnet mønster i data [58]. Figur 2.7 illustrerer et histogram med 30 bins.



Figur 2.7: Denne illustrasjonen viser et histogram som presenterer 30 søyleintervaller.

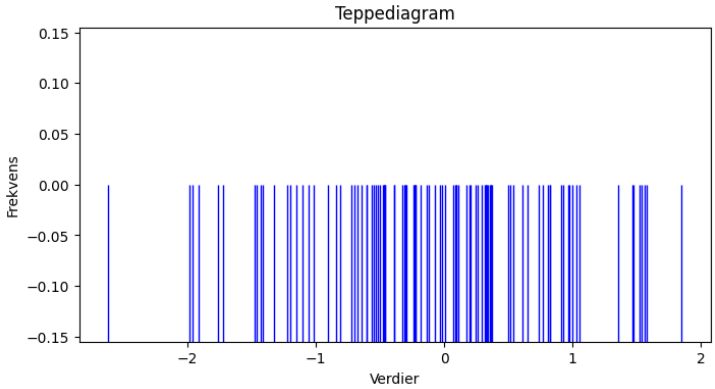
Fiolinplott

Fiolinplott (*eng. Violinplot*) illustrerer fordelingen av numerisk data. Fiolinens form og størrelse forklarer spredning av datapunkter, der bredden viser til frekvensen av data av ulike verdier. Fiolinplott gir en tydelig tetthetskurve som forklarer variasjonen i datapunktene. I tillegg visualiseres det andre nøkkeltall som median, kvartiler og halehenger (*eng. Whiskers*). Kombinasjonen av disse nøkkeltallene og tetthetskurven oppsummerer den statistiske sammenhengen i datasettet, samtidig som den viser fordelingen av datapunktene i forhold til tettheten. Fordelene ved fiolinplott er mest brukbar når flere utvalg av datasettet blir sammenlignet med hverandre [59]. Da viser den til både likheter og ulikheter til de fremhevede andelene. Figur 2.8 er et typisk eksempel på en fiolinplott.



Figur 2.8: Denne illustrasjonen viser en fiolinplott som presenterer 5 fioliner for separate variabler i et datasett. Fiolinens form representerer tettheten i datapunktene for hver variabel. I tillegg er det en forminsket boksplott inni fiolinen som presenterer median, første og tredje kvartil, samt halehengere.

Teppediagram (*eng. Rugplot*) en type graf som viser fordelingen av individuelle datapunkter langs en en-dimensjonal tallinje [60]. Hvert datapunkt er representert som en strek eller en prikk på linja og er spredt i forhold til en gitt fordeling. Det kan være krevende å tolke fordelingen av data ved de tilfellene der flere datapunkter overlapper hverandre på én linje. Imidlertid er det tydeligere å identifisere datapunkter som skiller seg fra de resterende. I kombinasjon med andre grafer som fiolinplott, er teppediagram er verktøy for å detektere utliggere som ligger lenger unna den generelle fordelingen. I Figur 2.8, blir det vist et teppediagram som er integrert med en fiolinplott lang y-aksen. Figur 2.9 viser et annet eksempel på et frittstående teppediagram.

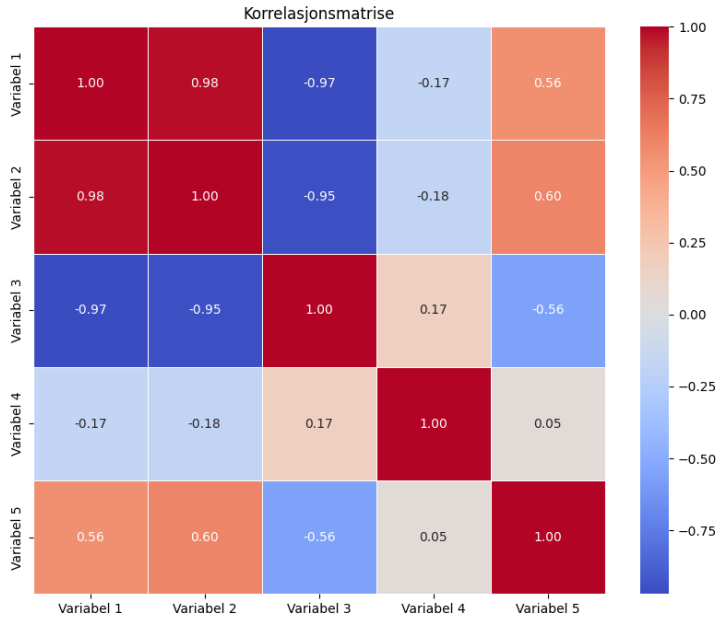


Figur 2.9: Denne illustrasjonen viser et teppediagram som beskriver fordelingen av datapunkter i en variabel. Punktene representerer Alle strekene representerer et datapunkt fra denne variabelen.

Korrelasjonsmatrise

En korrelasjonsmatrise gir en visuell oversikt over hvordan flere variabler i et datasett korrelerer med hverandre. Matrisen består av koeffisienter som beskriver sammenhengen mellom alle kombinasjoner av variablene. Visualisering av en slik matrise kan benyttes til å undersøke hvilke variabler som er mer avhengige og uavhengige av hverandre [17].

Pearsons korrelasjonskoeffisient brukes for gi en indikasjon på hvilken grad variablene henger sammen. Koeffisienten er ofte fremmet ved variabelnavn r, der intervallet på variabelen er $[-1, 1]$ [61]. De tilfellene der verdien er -1 viser det en perfekt lineær sammenheng som er negativ. Dette er et teoretisk tilfelle, og det samme vil gjelde når den er positiv, da $+1$ indikerer på en lineær sammenheng som er perfekt. En tallverdi på 0, betyr det at det ikke er noen form for sammenheng mellom variablene. Generelt sett vil $|r| > 0.5$ indikere på at variablene er sterkt korrelerte og viser til en robust lineær sammenheng. Når $0.3 < |r| < 0.5$ viser det til noe merkbar sammenheng mellom variablene. $0 < |r| < 0.3$ viser det svake korrelasjoner som ikke har stor betydning for endring i datapunktene.Begrenset antall variabler muliggjør visuell inspeksjon av korrelasjoner mellom variablene med konstruksjon av korrelasjonsmatrise. Matrisen har variabler langs begge aksene, der krysningen mellom dem viser en koeffisient for hvor grad de er korrelerte. Figur 2.10 er et eksempel på en typisk korrelasjonsmatrise.



Figur 2.10: Denne illustrasjonen viser en korrelasjonsmatrise som viser hvor mye hver av variablene korrelerer med hverandre.

Autokorrelasjonsfunksjon plott - ACF plott

ACF-plott visualiserer autokorrelasjon mellom etterfølgende observasjoner i, med fast tidsintervall. Autokorrelasjon beskriver korrelasjonen mellom en observasjonen og en tidligere observasjonen. Ved en slik analyse er det mulig å se nye sammenhenger i data, med hensyn på tid. Autokorrelasjonsfunksjon plott (*eng. ACF-plot*) er en av flere teknikker som brukes for å finne autokorrelasjoner [56].

2.2 Algoritmer for regresjon

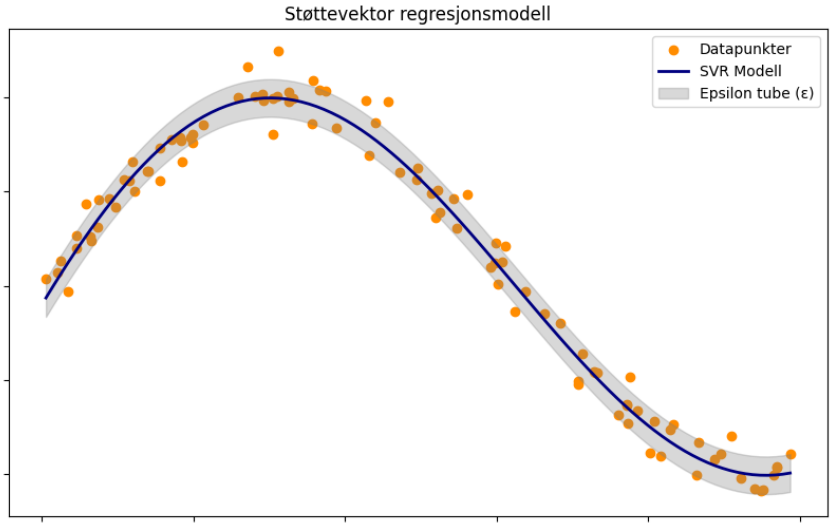
Regresjon er en metode innen statistikk som analyserer relasjoner mellom variablene. Hensikten er å forstå sammenhengen mellom prediktorer og kontinuerlige responsvariabler. Denne innsikten er kritisk for å predikere verdier innen kontinuerlig data. Når denne metoden benyttes er det mulig å klargjøre forholdet mellom avhengige og uavhengige variabler i tillegg til at den kan identifisere trender og mønstre [62]. I følgende seksjon vil det presenteres for ulike algoritmer inn veiledede regresjonsmetoder og semi-veilede regresjonsmetoder.

2.2.1 Klassiske veiledede algoritmer for regresjon

Veiledet regresjonsmodeller er en av to kategorier innen veiledet maskinlæring. Disse algoritmene trenes på datasett som inneholder både uavhengige og avhengige variabler. Dette gjøres for å kunne forutsi utfallet for ny, usett data. Innen dette feltet er det mange algoritmer som løser ulike typer problemer. Støttevektor regresjonsmodeller (SVR) er til for marginbaserte tilnærminger. Random Forest Regressor brukes for beslutningsmodellering og er inspirert av algoritmen beslutningstrær. K-nærmeste naboer regresjon (KNR) brukes for instansbasert læring.

SVR - Støttevektor regresjonsmodell

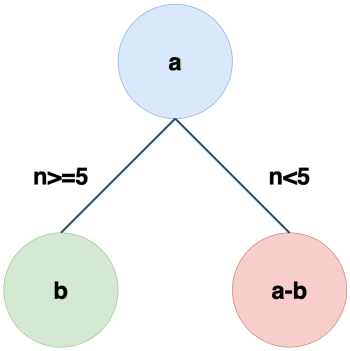
Støttevektor regresjonsmodellen (SVR) (*eng. Support Vector Regressor*) er en veiledet maskinlæringsalgoritme. Dette er en versjon av støttevektor klassifikasjonsmodell (SVM) (*eng. Support Vector Machine*) som er rettet mot å løse regresjonsproblemer. Når SVR blir brukt for predikere, blir det satt en terskel rundt selve modellen. Denne terskelen justeres ut ifra parameteren epsilon (ϵ) [63]. Denne setter en grense og lager en tube rundt modellen som vist i Figur 2.11. Dette blir sett på som marginfeil (*eng. Margin of Error*) der feilen til datapunktene innenfor denne grensen blir ignorert og vurdert som "tillatt", mens datapunktene utenfor tuben blir sett på som utliggere. Når det er utliggere utenfor tuben regnes avstanden mellom de, og brukes brukt for å justere (ϵ)-tuben.



Figur 2.11: Figuren ble konstruert i Python. Denne illustrasjonen viser en graf av hvordan SVR predikerer på gitte datapunkter. Verdien til epsilon (ϵ) definerer størrelsen på tuben rundt modellen.

Beslutningstrær

Et beslutningstre er en maskinlæringsmodell som har en struktur som gjenspeiler et tre. Hver grein har enkelte kriterier for å skille verdiene i datasettet. Målet er å få en homogen fordeling, der datapunktene som er mest lik hverandre fordeles til samme node. Dermed er viktig å vurdere hvilke kriterier som må settes for at informasjonsfordelingen blir optimal. Figur 2.12 viser et eksempel på hvordan informasjonsflyten mellom nodene i et beslutningstre overføres. Noden som overfører informasjonen kalles rotnode, mens nodene som tar imot denne informasjonen kalles bladnode.



Figur 2.12: Figuren illustrerer strukturen til et beslutningstre. Rotnoden er blå, mens de grønne og røde ansees som bladnodene. Figuren viser et eksempel på hvordan algoritmen fordeler datapunktene til sine respektive noder ved et gitt kriterium. Følgende kriterie fordeler alle verdier fra og med til den grønne bladnoden, som har b antall verdier. Alle verdier under 5 vil da bli sendt til den rød bladnoden, også sett på som differansen mellom rotnoden og den grænne bladnoden.

For å splitte opp datapunktene er det viktig å prioritere den variabelen som påvirker prediksjonsevnen til modellen [64]. Når den er funnet deles informasjonen opp til mindre andeler av data. Det finnes mange metoder for å finne den ideelle variabelen som bør splittes først. En mulighet er utførelse av prediktive analyser som "variabel viktighet" (*eng. Feature Importance*) på hele datasettet, for å finne hvilken variabel som påvirker modellytelsen [65].

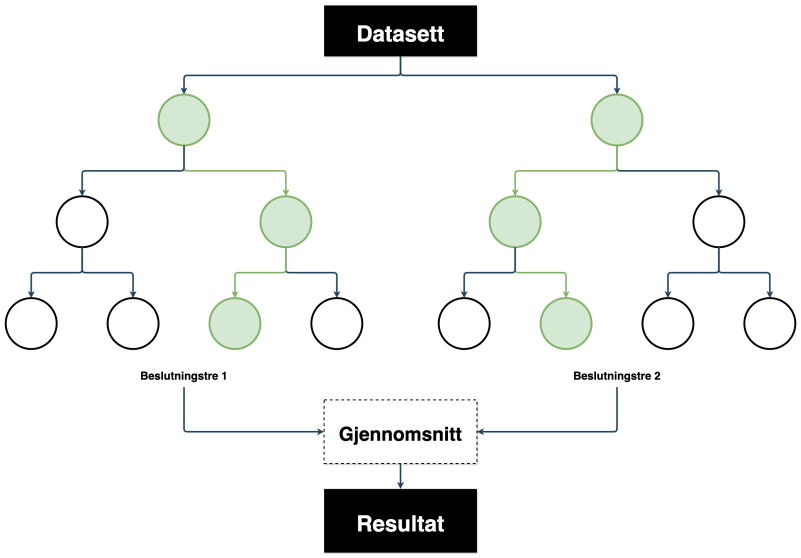
For å estimere graden av nøyaktighet i fordelingen av datapunktene, regnes det et "urenhetensmål" (*eng. Impurity Measure*) på splittingen. Denne målingen blir gjort ved å bruke evalueringsmetrikken RMSE (se Seksjon 2.1.4). Det kalkuleres en RMSE-verdi for rotnoden og begge bladnodene. Deretter ser man på differansen i RMSE-verdiene mellom rotnoden og de vektede RMSE-verdiene for begge bladnodene, hver for seg. Deretter velges de nodene med størst differansene. Dette vil gi høyest varians reduksjon og forklarer informasjonsfordelingen ideelt. Forskning av Chokka A. (2019) [66], viser til at det er mer effektivt å regne evalueringsmetrikken RMSE, istedenfor å bruke standardavviket i nodene. Dersom det viser seg at prediksjonsevnen til modellen er svak og RMSE-verdien blir for høy vil modellen prøve seg fram medandre kriterier fram til den får en optimal fordelingen av data.

RandomForestRegressor

Random forest regressor (RFR) er en tre-basert algoritme innen veiledet læring som baserer seg på beslutningstrær. Algoritmen er en robust modell som ofte får gode resultater sammenliknet med andre maskinlæringsalgoritmer. Random forest regressor er bygd opp der den kombinerer flere enkle beslutningstrær som vist i Figur 2.13. Modellen kommer fram til en prediksjon ved å ta gjennomsnittsverdien for individuelle beslutningstrær når den løser regresjonsproblemer. Når flere beslutningstrær blir kombinert vil de enkle trærne unngå å bli utsatt for høy varians [17]. Denne kombinasjonen introduserer mer diversitet i modellen og gjør den mer robust. Teorien bak dette er inspirert av ensemble læring, der hovedfokuset er å øke treffsikkerheten og kontrollere overtilpasningen til en modell [67].

Ensemble læring en samling av flere enkle maskinlæringsalgoritmer. Her blir det valgt ut prediksjoner som har blitt generert av flere modeller. Dette blir gjort for å forbedre treffsikkerheten og prediksjonsevnen til modellen [68]. *Bagging* er en teknikk innen ensemble learning. Når bagging anvendes vil det være et sett med flere enkle maskinlæringsmodeller som blir med på å avgjøre det endelige resultatet [69]. Hver modell tar ut et tilfeldig utvalg av hele datasettet. Disse utvalgene er unike for hver modell og de trenes hver for seg [70]. Når resultatene hentes ut etter treningen, ser man etter flertallet av de modellene med likt utfall. Dette forbedrer evnen til å generalisere modellen for flere tilfeller [71].

En av grunnene til at random forest er populær er fordi den er tilpasningsdyktig for flere typer problemer. Algoritmen kan takle datasett med nokså høy dimensjonalitet og kompleks data. Random forest tar ut en tilfeldig andel av datasettet og trener det opp ved å sette visse begrensinger for å sortere datapunktene. I tillegg har algoritmen en tendens til å få gode prediksjoner selv når det er begrenset med data [72].



Figur 2.13: Figuren illustrerer fordelingen av trærne i algoritmen RandomForest. Det blir vist to beslutningstrær som viser hvilke noder som blir valgt i grønn farge. Det endelige resultatet på prediksjon er gjennomsnittsverdien av de predikerte verdiene til beslutningstrærene.

KNR - K-Nærmeste Nabo Regresjon

K-Nærmeste Nabo Regresjon (KNR) er en veiledet maskinlæringsalgoritme som løser regresjonsproblemer. Algoritmen er inspirert av klassifikasjons varianten K-Nærmeste Nabo Klassifikasjonsmodell (KNN) [73]. KNR er kjent som en lat algoritme, da den ikke trener modellen på samme måte som andre algoritmer. I motsetning til SVR og RFR, er KNR intuitiv og gjør ingen antagelser om den underliggende fordelingen av datapunktene. Prediksjonen av verdien til et datapunkt baseres på de k-nærmeste datapunktene i treningssettet. Dermed er det ingen definert treningsfase for avdekking av mønstre og sammenhenger. Siden algoritmen tar en vurdering basert distansen til datapunktene, er den i tillegg stand til å løse ikke-lineære problemer [17].

Når verdien til et datapunkt predikeres, beregner algoritmen en avstand mellom gjeldende punkt og alle andre datapunkter i treningssettet. Dette gjelder for både klassifikasjon og regresjonsvarianten. Avstandsmetrikken kan variere og baseres på Minkowski formelen og er definert med variabelen p, som vist som i Formel 2.6. Ved tilfellene der $p = 2$ vil Minkowski formelen bli gjort om til en Euklidsk avstandsformel, som er vanlig å benytte i slike tilfeller [17]. Etter beregning av distanser til hvert datapunkt, utvelges de k-nærmeste datapunktene. Hvilke datapunkter som blir utvalgt som verdige naboer avhenger av hvilken avstandsmatrise som benyttes. Derfor er det flere måter å beregne prediksjonsverdien til det gjeldende datapunktet. Eksempelvis kan den endelige predikerte verdien være et gjennomsnitt av verdiene til de k-nærmeste identifiserte naboene. En alternativ måte kan være å ta en vektet gjennomsnitt fordi det gir større betydning til naboene basert på hvor nærmere de er [73].

$$d(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \left(\sum_k \left| x_k^{(i)} - x_k^{(j)} \right|^p \right)^{\frac{1}{p}} \tag{2.6}$$

2.2.2 Semi-veiledede algoritmer for regresjon

I dette studiet ble det benyttet 3 semi-veiledede regresjonsmodeller. Den første modellen kombinerer prinsippene fra selv-læring (Seksjon 2.1.2) med algoritmen RandomforestRegressor. Den andre modellen benytter seg av teorien fra med-læring (Seksjon 2.1.2) og bruker algoritmen KNR som et utgangspunkt. Den tredje modellen, BHD er hentet og inspirert fra biblioteket LAMDA (se Vedlegg A.3.2). Ettersom den den førstnevnte modellen er egendefinert, er den forklart i Metoden under Seksjon 4.3.1. Videre vil de to sistnevnte algoritmene forklares under.

COREG

COREG (*Regression with Co-Training*) er en semi-veiledet maskinlæringsalgoritme som baserer seg på teorien bak med-læring. Algoritmen er bygget opp av to separate KNR-algoritmer, som parallelt forsøker å finne en unik løsning på et regresjonsproblem. COREG benytter seg av semi-veiledet læring ved å la KNR-modellene ta vurdering basert på ulike deler av treningsdata, både markerte og umarkerte data. Hensikten med treningsfasen i COREG er å utvide vurderingsgrunnlaget til hver KNR-modell, ved å inkludere kvalifiserbare datapunkter som opprinnelig var umarkerte til pseudomarkerte datapunkter. De mest informative datapunktene som bidrar til å forbedre modellenes prediksjonsevne blir antatt som kvalifiserbare. For KNR-modellene er de datapunktene som ved inkludering med tilhørende pseudomarkering reduserer samlet prediksjonsfeil sammenlignet med ekskludering. Hvilke datapunkter som anses som kvalifiserbar kan variere mellom KNR-modellene. Treningsfasen til COREG går ut på at hver KNR-modell vurderer hvilke umarkerte datapunkter som anses som kvalifiserbare og deretter vurdere å inkludere det til hver sitt respektive treningssett. Dette skillet i informasjonen fører til at vurderingene er tatt fra ulike innfallsvinkler og kan videre bidra med å øke generaliserbarhet til COREG modellen [16]. Endelig prediksjon på usett data vil basere seg på den samlede vurderingen av de separate modellene.

Når COREG anvendes er den avhengig av å ha et sett med markerte datapunkter L og et sett med umarkerte datapunkter U. Begge KNR-modellene kan navngis som KNR_1 og KNR_2. Hver modell tildeles en versjon av det markerte settet, i form av L_1 og L_2. For å definere algoritmenes struktur og vilkår, er det nødvendig å presisere k-verdi og p-verdi for hver av modellene, som forklart i Seksjon 2.2.1 om KNR. Deretter defineres maksimalt antall iterasjoner T som skal gjennomføres, der hensikten er å sette en begrensing på læringsprosessen til modellen.

Under hver iterasjon velges det et tilfeldig utvalg U' av U. Hvert enkelt umarkert datapunkt x_u i utvalget blir tatt i betraktning og vurderes av hver KNR-modell. For hver x_u, lages det en alternativ versjon h' av gjeldende KNR-modell. h_1 og h_2 er basert på de respektive markerte settene L_1 og L_2 for øyeblikket. De alternative versjonene h'_1 og h'_2 baserer seg på de samme markerte settene, samt inkludering av x_u og tilhørende predikerte markering $\hat{y}_u$.

Hver modell vil ta en vurdering på om den presterer bedre eller mer konsist med inkludering av det umarkerte datapunktet. Prestasjonen er basert på prediksjonen av de k-nærmeste naboene til x_u (Ω). Det beregnes en "markerings-sikkerhet" Δ (*eng. Labeling*

Confidence), basert på differansen i prediksjon av Ω med og uten det umarkerte datapunktet for hver x_u. Dersom de samlede prediksjonene av Ω til h' er mer lik de faktiske markeringene til Ω sammenlignet med h, vurderes det umarkerte datapunktet som kvalifiserbar. Hvis modellen presterer dårligere til tross for dette, vurderes datapunktet som ikke kvalifiserbar og forblir i det umarkerte datasettet U.

Alle kvalifiserbare x_u med positiv "markerings-sikkerhet" ($\Delta > 0$) i utvalget U' vil samles og vurderes. Det datapunktet med høyest "markerings-sikkerhet", blir utvalgt av de respektive modellene. KNR_1 overfører det umarkerte datapunktet fra U til L_1 og KNR_2 fra U til L_2. I det tilfellet begge KNR-modellene vurderer det samme umarkerte datapunktet som mest egnet, velger KNR_2 den nest egnede. En slik tilnærming gjennomføres for å forbedre generaliserbarheten til modellen. Denne prosessen gjennomføres til maksimale iterasjoner T er nådd, eller når ingen av de gjenværende datapunktene vurderes som kvalifiserbare.

Ved prediksjon av usett data, vil hver KNR-modell predikere selvstendig. Resultatet av treningsfasen er KNR_1 og KNR_2 som baserer vurdering på ulik data. L_1 og L_2 dekker ulike deler av det fullstendige treningssettet. Prediksjonene fra hver modell vil dermed være unik. Endelig resultat vil være et gjennomsnitt av prediksjonene til de respektive modellene [16]. Slik beregnes et optimalt resultat basert på en samlet vurdering.

Nedenfor vises det til pseudokode av COREG-modellen, hentet fra følgende studiet [16].

Algorithm 1 COREG

Require: Markert datasett L, umarkert datasett U, antall nærmeste nabo k, maksimale iterasjoner i treningsfasen T, distansemetrikker p_1, p_2

Ensure: Regressor $h^*(x) \leftarrow \frac{1}{2}(h_1(x) + h_2(x))$

$L_1 \leftarrow L$; $L_2 \leftarrow L$
Lag tilfeldig utvalg U' fra U
$h_1 \leftarrow kNN(L_1, k, p_1)$; $h_2 \leftarrow kNN(L_2, k, p_2)$
repeat
    for $j \in \{1, 2\}$ **do**
        for each $x_u \in U'$ **do**
            $\hat{y}_u \leftarrow h_j(x_u)$
            $\Omega \leftarrow Neighbors(x_u, k, L_j)$
            $h'_j \leftarrow kNN(L_j \cup \{(x_u, \hat{y}_u)\}, k, p_j)$
            $\Delta_{x_u} \leftarrow \sum_{x_i \in \Omega}((y_i - h_j(x_i))^2 - (y_i - h'_j(x_i))^2)$
        end for
        if det eksisterer en $\Delta_{x_u} > 0$ **then**
            $\tilde{x}_j \leftarrow \arg\max \Delta_{x_u}$; $\tilde{y}_j \leftarrow h_j(\tilde{x}_j)$
            $\pi_j \leftarrow \{(\tilde{x}_j, \tilde{y}_j)\}$; $U' \leftarrow U' - \pi_j$
        else
            $\pi_j \leftarrow \emptyset$
        end if
    end for
    $L_1 \leftarrow L_1 \cup \pi_2$; $L_2 \leftarrow L_2 \cup \pi_1$
    if hvis verken L_1 eller L_2 endres **then**
        exit
    else
        $h_1 \leftarrow kNN(L_1, k, p_1)$; $h_2 \leftarrow kNN(L_2, k, p_2)$
        Lage et nytt tilfeldig utvalg U' fra U
    end if
until T runder

BHD - Varmediffusjon med grensebetingelser i graf

BHD er en semi-veiledet maskinlæringsalgoritme som baseres seg på teorien bak grafbasert-læring. Algoritmen benytter fysisk teori om varmediffusjon i et lukket system med grense-betingelser til å forbedre etikettpropagering [74]. Her sammenlignes flyt av varme mellom punkter i et nettverk med propagering av etikker mellom noder i en graf. Modellen be-nytter seg av semi-veiledet læring ved å bruke markerte data som grensebetingelser på umarkerte data ved propagering av etiketter. Hensikten med en slik tilnærming er at det kan bidra til å kontrollere etikett propagering i fordelaktig retning og mulig redusere feil i prediksjon i større grad.

Algoritmen består av flere komponenter og kan inndeles i flere stadier. Først konstrueres det en graf av fullstendig data fra forklaringsvariablene, uavhengig av markering. Grafen kan være fullstendig eller delvis knyttet, vektet eller ikke vektet. Deretter gjennomføres det særegne varmediffusjonen med grensebetingelser. Utførelsen kan inndeles i to stadi-er som kombineres. Basert på den konstruerte grafen, gjennomføres det først en vanlig etikettpropagering som beskrevet i Seksjon 2.1.2. Resultatet er at alle datapunkter er tildelt markeringer basert på informasjon fra det markerte delen av data, uten ytterligere begrensninger. I algoritmen omtales dette stadiet som beregning av $''harmonic\_score''$.

Deretter introduserer grensebetingelsen for umarkerte datapunkter. Det innebærer å til-dele markeringer på umarkerte data basert på statistisk informasjon om markerte data. Stadiet omtales som fastsettelse av "initiell temperatur". Det skal virke som et forut-bestemt utgangspunkt for markeringene av de umarkerte datapunktene basert på kjent informasjon. Eksempelvis kan det globale gjennomsnittet til kjente markeringer benyttes. I en iterativ prosess benyttes informasjonen fra *harmonic_score* og den "initielle tempe-raturen" til etikettpropagering som etterligner varmediffusjon. Under denne prosessen styres grad av etikettpropagering av en diffusjons-koeffisient α. Jo høyere koeffisient, desto større innflytelse har informasjonen fra nabonodene i grafen.

Nedenfor legges det til pseudokode av algoritmen, hentet fra gjeldende studiet [74]. I koden beskriver n antall observasjoner i datasett. I algoritmen er endelig estimat på *harmonic_score* betegnet som Y^K, og k beskriver k-te iterasjonen i beregning av det. Grensebetingelse er betegnet som konstant C. Prosessen som etterligner varmediffusjo-nen, er beskrevet i linje 12. Og endelige markeringer er betegnet som S.

Algorithm 2 BHD: Varmediffusjon med grensebetingelser i graf

Require: Datasett (str.: $n \times m$); Overgangsmatrise T (str.: $n \times n$); initiell markeringsvektor Y (str.: $n \times 1$); Laplacian matrise L (str.: $n \times n$); Identitetsmatrise I (str.: $n \times n$); Antall iterasjoner M (standardverdi: 30); Diffusjons-koeffisient α (parameterverdi: [0,1])

Ensure: Tilstand_Vektor S ($n \times 1$)

Initier $U \leftarrow Y$
repeat
    $Y_{k+1} \leftarrow TY_k$
    $Y_{k+1} \leftarrow Y_{k+1} + U$
    $Y_k \leftarrow Y_{k+1}$
    $k \leftarrow k + 1$
until feilmargin mellom Y_{k+1} and Y_k blir tilstrekkelig liten
Initiell_Temperatur: Estimer verdier for umarkerte noder basert på gjennomsnitt av markerte noder
$C \leftarrow$ Initiell_Temperatur $- Y^K$
S $\leftarrow C$
for $b = 1$ to M **do**
    S $\leftarrow Y_K + \left(I - \frac{\alpha}{M} L\right)$ S
end for
return S

2.3 CRISP-DM

CRISP-DM (Cross-Industry Standard Process for Data Mining) er en metodologi som tilbyr et standardisert rammeverk for å muliggjøre systematisk tilnærminger til datautvinningsprosjekter [75]. Metoden teknologi- og problemnøytral, og kan dermed benyttes uavhengig av prosjektets natur. Rammeverket settes ved å bygge en forståelse for hensikten til prosjektet fra et forretningsperspektiv. Gunstig anvendelse av metodologien sikrer dermed at datautvinningsprosjektet holder riktig kurs i forhold til definerte forretningsmål. Det forsikrer at analyser og resultater ikke avviker fra formålet. Metodologien består av seks sekvensielle faser:

1. Forretningsforståelse:

Innledende fase bidrar til å etablere et solid grunnlag for prosjektet fra et forretningsperspektiv. Det fokuseres på å forstå mål og krav til prosjektet basert på helhetlig forståelse av flere elementer. Nåværende situasjon, eventuelle utfordringer virksomheten kan møte på, forventede konsekvenser av prosjektet, suksesskriterier og antagelser er blant annet aspekter som bør tas hensyn til. Det muliggjør identifisering av type data, analyse og form for resultat nødvendig for å kunne løse problemet og stå i lag med hensikten til prosjektet.

2. Dataforståelse:

Fasen involverer innsamling, beskrivelse og undersøkelse av data, relevant i forhold til etablert forretningsforståelse. Det gjøres for å gå en grundig forståelse av innhold, struktur, mangler og utfordringer knyttet til gjeldende data. På den måten kan kvaliteten av data vurderes og mulige tiltak for behandling av utfordringer utforskes.

3. Dataforberedelse:

Behandling av data omhandler forbedring av gjeldende datasett basert på mangler og utfordringer identifisert. Fasen innebærer inkludering, ekskludering, rengjøring, konstruering, integrering og formatering av data. Resultatet vil være data i gunstig tilstand, klargjort for benyttelse til videre analyse og modellering.

4. Modellering:

I denne fasen utvikles og benyttes variasjon av modeller av forskjellige modelleringsteknikker på behandlet data. Det innebærer valg av relevante dataanalyseverktøy, algoritmer og teknikker, generering av test design, bygge og vurdering av modell. Resultatene fra modell skal vurderes i henhold til fagkunnskap, definerte suksesskriterier og test design.

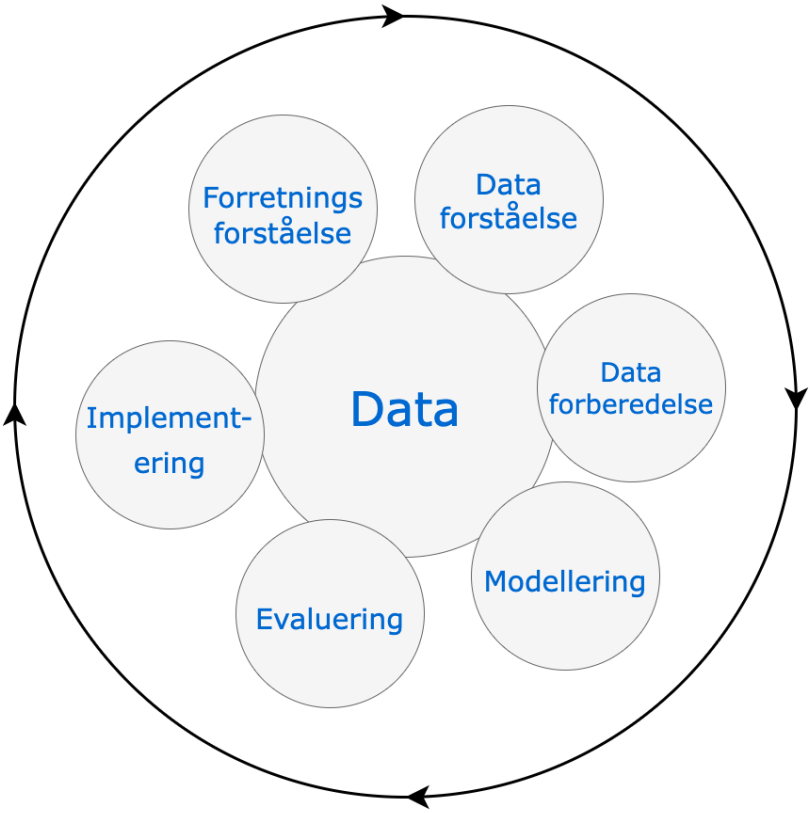
5. Evaluering:

Den utviklede modellen fra foregående fase, evalueres i forhold til hvordan de løser problemet. Dette innebærer tekniske vurderinger som modellens nøyaktighet, pålitelighet og effektivitet. I tillegg skal det gjøres forretningsmessig vurdering i form av evne til å møte definerte forretningsmål og suksesskriterier i tidligere fase.

6. Implementering:

Fasen innebærer implementering av den utviklede modellen etter teknisk og forretningsmessig evaluering til tilfredsstillende grad. Løsningen integreres i virksomheten. Det tilrettelegges ressurser til overvåkning og vedlikehold av modellen etter implementering. På den måten forsikres benyttelse av løsningen i en lenger tidshorisont.

Til denne oppgaven har CRISP-DM blitt benyttet for å etablere en strukturert framgangsmåte for blant annet utforskning, behandling og evaluering. Metodologien har bidratt til en standardisert flyt av prosesser, som er vært gunstig til forskningens formål.



Figur 2.14: Denne figuren viser en illustrasjon av rammeverket CRISP-DM. Her starter første steg ved forretningsforståelsen, og deretter følger man neste trinn retning med klokka.

2.4 MCDA

MCDA (*eng. Multi-Criteria Decision Analysis*) er et verktøy for flermålsanalyse og benyttes som metode for beslutningsstøtte for valg av ulike alternativer [76]. Evaluering av en situasjon med flere mål og kriterier i betraktning, kan være krevende. Metoden tilbyr en strukturert tilnærming til oversikt og vurdering av slike sammenfattende situasjoner, der flere aspekter skal tas hensyn til. MCDA bygger på antagelsen om at det er kan være flere relevante kriterier, men at de også kan eksistere på forskjellige skala [77]. Metoden sikrer dermed en samlet evaluering, basert på en vektet tilnærming til kriteriene. Slik virker verktøyet som støtte for beslutningstakere til mer fullstendig evaluering for valg av alternativer.

Analysen er bygget opp av en eller både kvalitativ og kvantitativ metode. Kvalitativ metode handler om vurderinger basert på sammenhenger i innsamling av empirisk data som ikke kan uttrukkes i numeriske from. Kvantitativ metode er basert på data i form av tallverdier [76]. Valg av metode avhenger av hvilke kriterier som alternativene skal vurderes etter og krav av type informasjon som kreves.

Hensikten med bruk av en MCDA-analyse i denne forskningen, er for å få en helhetlig evaluering av maskinlæring til prosesstyring og som beslutningsstøtte i produksjonssammenheng. Evalueringsmetrikker gir en teoretisk vurdering av algoritmenes prestasjoner for gitt datasettet. Imidlertid vil det nødvendigvis dekke mulige praktiske aspekter ved eventuell implementering. Da oppgaven undersøker flere muligheter og alternativer, bidrar rammeverket til en strukturert sammenligning og evaluering, basert på flere kriterier.

Kapittel 3

Materiale

Kapittelet beskriver hvilke materialer som ble benyttet til oppgavens formål. For å utforske mulighetene for å integrere en semi-veiledet regresjonsmodell i biokjemisk og industriell sammenheng, ble det benyttet et virkelig datasett fra en biokjemisk prosessproduksjon. Det vil gjøre rede for hvor datasettet er hentet fra og beskrivelse av produksjonen. Deretter vil metoden for datainnsamlingen og innholdet i datasettet forklares. Avslutningsvis blir det gjort rede for hvilke forbehandlinger som ble gjort i datasettet før gjennomførelsen av studiet.

3.1 Case fra bioprosessering

Bioco er et bioraffineringsanlegg som bruker enzymatisk hydrolyse til å oppgradere restråstoff fra fjørfe. Råstoffet er rester fra produksjon av kylling og kalkun, og består for det meste av skrog og skinn. I enzymatisk proteinhydrolyse kvernes råstoffet, før det blandes med vann og enzym og varmes opp til en bestemt temperatur [78]. Enzymene bryter ned store proteiner til mindre proteiner og peptider, som kan inngå som ingredienser i dyrefôr eller mat. I prosessen finnes det en rekke sensorer som måler viktige parametere som temperatur, trykk og strømningshastighet kontinuerlig. Under forskningsprosjektets periode var det i tillegg utplassert en NIR-sensor for spektroskopisk analyse av råvarematerialе. Deretter har det blitt utført laboratorieanalyser av ulike kvalitetsparametere på et begrenset antall produkter. Ettersom datasettet bestod av en stor mengde forklaringsvariabler fra kontinuerlige sensorer, og en begrenset mengde med markerte data (kvalitetsmålinger på sluttprodukt), var datasettet aktuelt og relevant til å benytte semi-veiledet regresjon.

3.1.1 Beskrivelse av prosessen

Anlegget kjøres døgnkontinuerlig og prosesserer omtrent 40 tonn råmateriale per dag. Blandingsforholdet mellom kylling og kalkun varierer og registreres ikke, og er dermed tolket som ukjent. Strømningshastighet, forhold mellom råmateriale og vann, og temperaturer er viktige prosessparametere som styres og måles kontinuerlig. Det brukes én enzymtype om gangen, med fast konsentrasjon. Sluttproduktet er en samling av én hel ukesproduksjon på 200 tonn råmateriale, og blir ansett som en batch. Når kvaliteten på produktet skal vurderes, vil det bli gjennomført basert på en samlet vurdering av batch-

en den tilhører. Prosessen stoppes opp på slutten av en arbeidsuke. Dette sørger for at utstyret i bioraffineringsanlegget klargjøres for ny oppstart til uken etter.

3.1.2 NIR

Nær infrarød spektroskopi (NIR) utgjør en analytisk metode innen matforskning, og brukes blant annet for å analysere innholdet i råstoffet. NIR-sensor tar nytte av egenskapene til infrarødstråling for måling av spektrale responser, ved å trenge seg gjennom ulike materialer og deretter måle mengden lys som absorberes. Den infrarøde strålingen er en type stråling som er på det elektromagnetiske spekteret og har en bølgelengde mellom 750 nm og 2500 nm [79].

Strålene fra NIR-sensoren måler sammensetningen av råstoffet, inkludert karbohydrater, fett, protein, vann og andre faktorer relatert til produktkvalitet [8]. Fordelen ved bruk av en NIR-sensor er evnen til å analysere produktet uten å påvirke produksjonsprosessen, og slipper mulig reduksjon i råmaterialets kvalitet og sikkerhet. Dette muliggjør kontinuerlig monitorering av råmateriale, samtidig som det samme produktet kan bli sendt ut til kunden [80]. I tillegg til at denne metoden er kostnadseffektiv, bidrar den til en bærekraftig løsning ved at det minimerer avfall. Når NIR-sensoren analyserer råmateriale, samler den raskt inn data som er relatert til næringsinnholdet i sluttproduktet som videre indikerer på produktkvalitet. Dette bidrar til å kunne effektivisere beslutningsprosesser og optimalisere produksjonsdriften.

3.1.3 Beskrivelse av datasettet

I regi av Bioco AS [81], Nofima [82] og forskningsprosjektet DigiFoods [83], ble det utført et stort industrielt forsøk for å utforske effekten av råstoff og enzymtyper på sluttproduktets egenskaper. Formålet var å finne ut hvordan ulike enzymtyper påvirket kvaliteten og smaken på produktet, for å utvikle et produkt som egner seg som ingrediens i mat. Produksjonen disse ukene avvek derfor fra standardprosessen, som hovedsakelig produserer ingredienser til dyrefôr.

Under forsøksperioden ble det samlet data fra utvalgte produksjonsuker mellom oktober 2022 og juni 2023. Forskjellige enzymtyper ble testet ukentlig i denne forsøksperioden. Første to dagene av uken var satt av til forsøk med én unik enzymtype og forskjellige råmaterialeblandinger. Samtidig som det ble utført kontrollerte endringer i tilsetning av råstoff og vannmengder. De resterende dagene var satt av til å utføre den vanlige standardproduksjonen med én fast enzymtype og ukjent blanding av råmateriale. Da formålet med forskningen var å undersøke hvordan de unike enzymene påvirket egenskapene i produktet, er den største andelen av laboratoriemålinger på kvalitet, fra produkter av den unike produksjonen under forsøkene.

Et annet mål med forsøket var å utforske hvordan spektroskopiske målinger av råmateriale kunne benyttes til å måle sammensetningen av råstoffet. For å måle sammensteningen i råstoffet ble det utplassert en NIR-sensor i startfasen av prosessen på Bioco. Sammensteningen som måles av denne sensoren er deriblant fordelingen mellom bein, proteiner og fett i blandingen. Data fra sensoren ble også samlet periodevis i nevnt tidsperiode for det

førstenevnte forskningsprosjekt. Datasettet har ingen registrerte verdier fra NIR-sensoren i uke 49. Resultater og data fra forsøkene var samlet inn til et datasett.

Datasettet var behandlet hos Nofima før det ble hentet for studiets undersøkelser. Grunnet konfidensialitet er faktiske enzymtyper anonymisert og erstattet med følgende enzymkoder; A1, A2, B, C, D og E. Det var lagt inn i beskrivelsen av datasettet at enzymkode "A2" beskrev enzymtypen som var benyttet i produksjonen utenfor forsøkene. Datasettet inneholdt også tidspunkt for datainnsamling. Tidspunktet beskriver tiden for en avgrenset mengde materiale inngikk i prosessen. Målinger gjort på gjeldende materiale ble justert for tid, slik at de sammenfalt med begynnelsen av prosessen. Eksempelvis var temperaturmåling på et stykke materiale 20 minutter inn i prosessen og kvalitetsmåling utlevert i ettertid av produksjonen, begge justert i datasettet slik at de var tilknyttet gjeldende materiale. Tidspunkt beskrevet i datasettet er dermed ikke forbundet med faktisk tidpunkt for de respektive målingene gjort underveis i prosessen. Verdienene i datasettet ble utsatt for en rekke behandlinger for å blant annet redusere støy. Alle datapunktene i de respektive målingene, ble utjevnet med et medianfilter med et tidsvindu på 30 minutter. Hvert datapunkt i datasettet er dermed et resultatet av de nærmeste 30 datapunktene med hensyn til tid.

Tabell 3.1 gir en oversikt over tidsperiodene for datainnsamlingen og informasjonen om enzymtypene, samt antall prøver som har gjennomgått kvalitetsundersøkelser for gjeldende periode. Tabell 3.2 gir en oversikt over aktuelle laboratoriemålinger og tilhørende beskrivelser. Tabell 3.3 gir en oversikt over sensormålinger med tilhørende beskrivelse. Følgende tre tabeller er lagt ved under.

Tabell 3.1: Oversikt over tidsperioder for datainnsamling, enzymtyper benyttet og antall prøver med kvalitetsmålinger for gjeldende periode. ***Uke*** 49* *har ingen tilgjengelige NIR-målinger.*

Uke (År):	**Dato:**	**Enzymkode:**	**Antall kvalitetsmålinger:**
44 (2022)	31/10 - 04/11	B/A2	114
45	07/11 - 11/11	C/A2	99
47	21/11 - 25/11	A1/A2	40
48	28/11 - 03/12	D/A2	75
49*	05/12 - 08/12	A1/A2	72
50	12/12 - 15/12	A2	64
24 (2023)	12/06 - 13/06	E	72

Tabell 3.2: Oversikt over laboratoriemålinger i datasett og tilhørende beskrivelse.

Kvalitetsmåling:	**Beskrivelse:**
Mw	Gjennomsnittlig molekylær vekt.
Small Molecules	Andel små molekyler.
Brix Adjusted	Brix fordelt på råmaterialeprosent.

Tabell 3.3: Oversikt over resterende variabler i datasett og tilhørende beskrivelse.

Variabelnavn:	**Beskrivelse:**
EnzymCode	Type enzym benyttet i prosessen
RawMatFlow	Råmateriale strømningshastighet
WaterFlow	Vannmasse flyt
RawMatPercent	Prosent av råmateriale i totalstrøm
RawMaterialMix	Type blanding av råmateriale
NIRfat	Andel fett i råmateriale
NIRprotein	Andel protein i råmateriale
NIRash	Andel aske i råmateriale
NIRwater	Andel vann i råmateriale
TT07	Temperaturmåling
TT08	Temperaturmåling
PT03	Trykkmåling
TT20	Temperaturmåling
TT09	Temperaturmåling
TT12	Temperaturmåling

Kapittel 4

Metode

4.1 Dataforståelse

Dataforståelse handler om innsamling av data relevant for prosjektet, i tillegg til å oppnå innsikt og forståelse om informasjonen [75]. Trinnet er kritisk og forståelsen av datasettet gir analytikerne oversikt som legger grunnlaget for videre analyser. Dette innebærer blant annet identifisering av avvik og utfordringer knyttet til datasettet. Videre vil det involvere fordeling og utforming av mulige datatyper og kategorier, samt mulige grupperinger av data. En grundig vurdering av slik informasjon vil bidra til verdifull innsikt om nødvendige tiltak i forbehandling av data. Analysen bidrar til å vurdere kvaliteten på data i form av relevans, kompleksitet og konsistens av informasjonen i datamengden. Informasjonen hentet ut i denne fasen vil ha betydningsfull innvirkning på hvordan data kan benyttes i modelleringsfasen.

Datasettet som oppgaven tok utgangspunkt i hadde en dimensjon på (43 251, 18) bestående av 15 forklaringsvariabler og 3 responsvariabler. Innholdet kan inndeles i markerte og umarkerte data, basert på observasjonene var knyttet til faktiske verdier i responsene. Det må presiseres om at alle markerte observasjoner inneholder faktiske verdier for alle kvalitetsmålingene som vist i Tabell 4.1. En markert observasjon har dermed faktiske verdier i alle respektive responsvariabler. Umarkerte observasjoner har ingen.

Majoriteten av datasettet bestod av umarkerte data, mens resterende markerte data utgjorde en svært liten andel. Tabell 4.2 viser blant annet oversikt over antall referanserverdier per kvalitetsmåling i hele datasettet. Med 540 faktiske verdier for hver kvalitetsmåling, var omtrent 98.6 % av data i hele datasettet umarkerte.

Semi-veiledede algoritmer tar utgangspunkt i informasjonen fra både markerte og umarkerte data, dermed var det nødvendig med separate undersøkelser for begge typer data i datasettet. Det var behov for å undersøke informasjonen i den umarkerte andelen av datasettet for å forstå hvordan informasjonen i de markerte og umarkerte delene av datasettet forholder seg til hverandre. Innholdet i umarkerte data gir innsikt i hvilken type informasjon de semi-veiledede modellene kan utnytte og hvordan det kan påvirke modellens forståelse av informasjonen.

Tabell 4.1: Oversikt over responsvariabler i datasettet og deres respektive beskrivelse.

Responsvariabel:	**Beskrivelse:**
Mw	Gjennomsnittlig molekylær vekt.
Small Molecules	Andel små molekyler.
Brix Adjusted	Brix fordelt på råmaterialeprosent.

Datasettet innholder informasjon om forskjellige faser i den industrielle produksjonsprosessen og kunne derfor inndeles i to kategorier. Som nevnt i Seksjon 3.1.3, bestod tidsperioden av datainnsamling for både standardproduksjon og produksjon under de ulike testforsøk. Det ble utført undersøkelser på hvordan forskjellige typer og mengder av enzymer og råmaterialeblandinger, påvirket kvaliteten og egenskapene på sluttproduktet fra produksjonen.

Produksjonen under disse testforsøkene avvek fra prosessene under standardproduksjonen med enzymtype "A2" og ukjent blandingsforhold i råmaterialeblandingen. Av den grunn kan data inndeles etter hvilken type produksjon de ble observert og samlet under. Observasjonene som representerer data fra forsøk med enzymtyper enn standardenzymet "A2" blir omtalt som deltsettet "Design"-sett. Designsettet beskriver de observasjonene av hele datasettet som tilhører produksjonen som var designet for testforsøkene. De øvrige observasjonene i hele datasettet som representerer standard produksjon med standardenzymet "A2", "Normal"-sett.

Tabell 4.2 viser en oversikt over størrelsen på de ulike delsettene. Det må understrekes at delsettene er en del av det hele datasettet og ikke er eget og separat datasett. Designsettet utgjorde mindre enn 35 % av datasettet, men inneholdt nærmere 67,4 % av de totale markerte observasjonene. Majoriteten av informasjonen i umarkerte data fra hele datasettet har opphav fra standardproduksjonen. Imidlertid inneholder normalsettet minder andel av de markerte data. En slik skjevfordeling av informasjon kan ha en betydningsfull innvirking på hva og hvordan semi-veiledede modeller lærer.

Tabell A.8, Tabell A.9 og Tabell A.10 i Vedlegg A.6, viser hvordan den gjennomsnittlige verdien av de ulike kvalitetsmålingene varierer utifra kombinasjonen av enzymtype og råmaterialblanding. For kvalitetsmålingen Mw, er det tydelig at enzymtype spiller en stor rolle for utbytte i målingene.

Tabell 4.2: Oversikt over dimensjoner, enzymkoder i prosessen og antall kvalitetsmålinger for hele datasettet, designsettet og normalsettet.

Datasett:	**Dimensjon:**	**Enzymkode(r) i prosessen:**	**Markeringer (Kvalitetsmålinger):**
Hele datasett	(43 251, 18)	Alle	540
Designsett	(14 038, 18)	A1, B, C, D, E	364
Normalsett	(29 213, 18)	A2	176

Forekomsten av manglende verdier varierte mellom de ulike variablene. Blant forklaringsvariablene hadde "TT12" lavest antall mangelende data (15), som tilsvarte omtrent 0.04 % av informasjonen i variabelen. "NIRprotein" hadde flest manglende verdier, med en verdi på (15 731), som utgjorde 36.37 %. Blant data i forklaringsvariablene var 9 % var manglende verdier. I kvalitetsmålingene utgjorde manglende data nærmest 99 %.

Datasettet bestod av både numeriske og kategoriske forklaringsvariabler. "EnzymCode" og "RawMaterialMix" var de kategoriske variablene. Enzymtype "A2" og råmaterialeblanding "Unknown mix" var de mest observerte typene i sine respektive kategoriske variabler, som vist i Tabell 4.3 og Tabell 4.4. De resterende numeriske forklaringsvariablene har blitt klassifisert i tre hovedgrupper som følgende; råmateriale- og vanninnstrømning, NIR-målinger, og temperatur- og trykkmålinger.

1. Råmateriale- og vanninnstrømning: "RawMatFlow", "WaterFlow" og "RawMatPercent" representerer informasjonen om strømningen og mengdenav råmateriale og vann i begynnelsen av prosessen.

2. NIR-målinger: "NIRfat", "NIRprotein", "NIRash" og "NIRwater" representerer målinger av fett, protein, aske og vann i råmateriale av nær-infrarød spektroskopi 3.1.2.

3. Temperatur- og trykkmålinger: Resterende variabler ("TT07" til "TT12") er industrielle målinger av trykk og temperatur av ulike deler under produksjonen.

Tabell 4.3: Oversikt over enzymtypene og tilhørende fordeling i hele datasettet. Tabellen viser at enzymtype "A2" utgjør den største delen av informasjonen blant enzymtypene, etterfulgt av enzymtype "A1".

Type:	**Fordeling:**
A2	67.54 %
A1	12.67 %
B	5.25 %
C	4.81 %
D	5.25 %
E	4.47 %

Tabell 4.4: Oversikt over råmaterialetypene og tilhørende fordeling i hele datasettet. "Unkown mix" representerer et ukjent blandingsforhold til råmaterialeblandingen og utgjør omtrent hele informasjonen i datasettet.

Type:	**Fordeling:**
Kylling	0.67 %
Kalkun	0.82 %
Ukjent blanding	98.51 %

Vurdering av tidsinformasjon

Ettersom dato og tid har blitt registrert under innsamling av data, kan datasettet karakteriseres som tidsseriedata. Tidserie er en liste med målinger, med informasjon om når målingene ble blitt registrert [84]. Tidsinformasjonen beskriver igangsettelsen til en spesifikk mengde råmateriale i prosessen. Datainnsamlingen fra sensormålingene har blitt registrert med et intervall på 1 minutt. Dette resulterte i at store deler av datasettet bestod av kontinuerlige sekvenser av observasjoner som var etterfølgende i prosessen. Det var flere tilfeller der det var avvik på tidsdifferansen mellom to etterfølgende observasjoner fra det faste intervallet.

Mangel på kontinuitet kan skyldes flere faktorer, som blant annet nedetid på sensor og stopp i produksjonen. Datasettet inneholdt 39 kontinuerlige sekvenser med fast tidsintervall på 1 minutt, med varierende varighet. Lengste sammenhengende sekvens av observasjoner var på 3 dager, 8 timer og 13 minutter. Sett bort fra mangel på datainnsamling utenfor produksjonstid,var lengste tidsintervall mellom to observasjoner på 6 timer og 30 minutter. Da er det også sett bort ifra overgangen fra uke 50 i 2022 til uke 24 i 2023, som vist i Tabell 3.1. Etter fast intervall på 1 minutt, var det også observasjoner med intervall på 2 minutter, etterfulgt av 1 time og 42 minutter. Med fravær av et fast tidsintervall mellom observasjonene, kan datasettet som tidsseriedata, beskrives som irregulært [84]. Irregulær tidsseriedata har varierende tidsintervaller mellom etterfølgende observasjoner i tid, og må som oftest behandles med fast intervall for å dra nytte av modeller som drar

fordel av informasjon som ligger i tid.

4.1.1 Visualisering av data

Ulike teknikker for visualisering kan bidra til å forstå kvaliteten på datapunktene. Når disse teknikkene implementeres kan man enkelt se fordelingen i data og få inntrykk av områder med mulige ekstrem data og unike karakteristikker [85]. Dette bidrar til å kunne avdekke eventuelle problemområder som må tas hensyn til eller behandles underveis.

I industrielle prosesser kan informasjonen i data skilles mellom sensormålinger av kontrollerbare styringsparametere og ukontrollerbare observasjoner. Kontrollmålinger som strømning og temperatur vil ha en egen fordeling ut ifra hvilke parameterverdier som blir satt under produksjonen. Siden disse parameterverdiene kan justeres etter behov, vil fordelingen og variasjonen i data være basert på disse verdiene. Innholdet i råmateriale i begynnelsen av prosessen, vil derimot ikke ansees som en kontrollerbar parameter. Konsekvensen av type variabel, vil påvirke fordelingen i data og utformingen når det visualiseres.

Ladningsplott fra PCA-analyse ble benyttet for å visualisere og forstå hvordan forklaringen av variasjonen i data var fordelt mellom de ulike variablene i datasettet. Siden variasjonen i informasjon spiller en stor rolle i hvordan en modell forklarer informasjonen i responsen, ble ladningsplottet benyttet med formål for å skille mellom mer viktige og mindre viktige variabler. Det var også benyttet til å forstå hvordan inkludering av NIR-data ville endre hvordan den totale variasjonen vil forklares mellom variablene. Siden verdier i ulike variabler kan skille i størrelsesorden, var det nødvendig å standardisere innholdet i datasettet før utførelse av PCA-analyse.

Histogram ble hovedsakelig benyttet for å undersøke hvordan fordelingen av data i hele datasettet var, samt forskjellige delsett. Det var benyttet for å se hvordan fordelinger i designsettet var sammenlignet med hele datasettet, og hvordan markerte og umarkerte data var fordelt i de respektive delsettene. Videre ble det benyttet for å se hvordan fordelingen data i variablene kunne grupperes etter de kategoriske variablene. På den måten var det mulig å undersøke i hvor stor grad informasjon i de kategoriske variablene kunne benyttes til å skille informasjonen i responsvariablene.

Fiolinplott viser også fordelingen av data, men hadde sin hensikt å undersøke og kartlegge mulig ekstrem data. I denne sammenheng var hele datasettet inndelt i markert, umarkert, normalsett og designsett, for å bli inspisert separat. En slik oppldeling muliggjorde å identifisere opphavet til datapunktene som kunne anses som ekstrem data. I tillegg var det gjort av nødvendighet slik at datapunktene ble vurdert etter riktig kontekst.

ACF-plott ble benyttet til å visualisere autokorrelasjonen mellom etterfølgende observasjoner for hver variabel. Informasjonen fra visualiseringen ble benyttet som terskelverdier for flere behandlingsmetoder utført på datasettet. Eksempelvis var resultater fra undersøkelsene benyttet til behandling av manglende verdier i datasettet, som blir videre forklart i Seksjon 4.2.2.

4.2 Dataforberedelse

Dataforberedelse og preprosessering av data utgjør en betydningsfull rolle i kvalitetsforbedring og organisering av data før modellering [75]. Behandling av data er nødvendig for at det skal kunne benyttes til modellering og implementeringsformål. Grundig og gjennomtenkt gjennomførelse av denne fasen sikrer at data blir behandlet til en tilstand som fører til effektiv behandlingskraft og tid for modeller. Det har en sterk påvirkning på prediksjonsytelsen, da det er med å utforme informasjon og detaljer i data. Fasen innebærer blant annet utvelgelse av hvilke variabler som skal inkluderes videre i prosessen og behandling av kategoriske variabler, samt håndtering av manglende verdier og ekstremverdier.

4.2.1 Utvelgelse av variabler

Valg av variabler er det første steget i dataforberedelsen, og innebærer undersøkelser av hvorvidt de påvirker datakvaliteten og modelltreningen. Valg av å beholde, eliminere eller behandle variabler vil ha betydningsfull innvirkning på evnen til modellen for å fange sammenhenger i datasettet. Det vil påvirke presisjon og hastighet på utvikling av modellen, innvirkning på overtilpasning og generaliserbarheten av modellen. Hvilke teknikker og beslutninger som utføres er avhengig av egenskapene og strukturen til datasettet. Det være ønskelig med variabler som er informative og tolkbare, da hensikten bak modellering er optimal prediksjonsytelse og forståelse av signifikansen til parameterene den består av. I betraktning av at datasettet består av mindre antall forklaringsvariabler, var det nødvendig med grundigere vurdering av håndtering av data. I den sammenheng var domenekunnskap om prosessen essensiell for endelig vurdering.

Lite korrelerte forklaringsvariabler er ønskelig, da det fører til mer forutsigbarhet og konsistent modellering. Når to eller flere forklaringsvariabler er sterkt korrelerte, er det fare for at det oppstår "multikollinearitet" (*eng. multicollinearity*). Det vil indikere mangel på uavhengighet mellom variablene. Multikollinearitet kan være problematisk ved modellering. Når flere variabler kan forklare samme type informasjon, skaper det usikkerhet bak hvordan modellen har forstått betydningene av de ulike variablene. Muligheten for at modeller kan forklare samme informasjon på ulike måter, bidrar til inkonsistens, uforutsigbarhet og mulig "bias" i resultater [86].

Multikollinearitet kan håndteres ved å enten ekskludere en eller flere av de høyt korrerte variablene. Et annet tiltak er å kombinere variablene, eksempelvis gjennom dimensjonsreduksjon med PCA-analyse (som beskrevet i Seksjon 2.1.2) . Imidlertid kan dimensjonreduksjon med PCA føre til tap av informasjon eller gjøre det krevende å tolke i forhold til de opprinnelige variablene. Ved ønske om tolkning av hvordan resultatet fra modellingene er påvirket av variablene, ble det valgt å ekskludere mindre viktige variabler som korrelerte sterkt, manuelt.

Med en korrelasjonsmatrise var det mulig å få oversikt over hvordan og i hvor stor grad hver variabel i datasettet korrelerte med hverandre. Par av variabler med koeffisient i absoluttverdi over 0.5, ble satt til vurdering for ekskludering fra datasettet. Resultatet fra analysen alene er ikke holdbart, og ga kun en indikasjon på hvilke par av variabler som måtte behandles. Det vil eksempelvis ikke kunne gi informasjon om hvilken spesifikk

variabel som eventuelt bør ekskluderes. Resultatene av utførte korrelasjonsmatrisen er lagt til i Vedlegg som Figur A.4.1.

Korrelasjon med responsvariabler skaper derimot ikke "multikollinearitet", og gjelder kun mellom forklaringsvariabler. Siden datasettet inneholdt flere responsvariabler, var det undersøkt for hvordan hver forklaringsvariabel korrelerte med hver responsvariabel. Tabell 4.5 viser et selektivt utsnitt fra korrelasjonsmatrise av variablene. Eksempelvis var "NIRfat" sterkt korrelert med respons "Smallmolecules" og lavt korrelert med respons "BrixAdjusted". En slik analyse kan ikke benyttes som definitive resultater, men gir en indikasjon på nødvendigheten av egen modellering for hver responsvariabel. Dersom det hadde vist seg at forklaringsvariablene oppførte seg på samme måte for hver respons og at hver respons oppførte seg lignende, ville det redusert behovet for å behandle dem som unike utfall.

Domenekunnskap kan gi innsikt om variablene og bidra til et bedre beslutningsgrunnlag for valg av variabler. Resultatet fra den visuelle inspeksjonen av korrelasjonsmatrisen ble dermed sett i sammenheng med tilegnet domenekunnskap om produksjonen datasettet beskrev. Slik var det mulig å avgjøre hvilke av de høyt korrelerte variablene som eventuelt kunne ekskluderes.

Eksempelvis var "TT09" og "TT20" sterkt korrelerte forklaringsvariabler. Det kan skyldes av at de respektive temperaturmålingene følger tett inntil hverandre i produksjonslinjen, som vist i visualiseringen av prosessen i Seksjon 3.1.1. Siden en eventuell implementering av modell er avhengig av innhenting av sanntidsdata, vil det være fordelaktig å ha med data som tilgjengeliggjøres tidligere i prosessen. Slik informasjon var benyttet til å styrke eksempelvis argumentasjonen om å velge "TT20" framfor "TT09" for å redusere antall sterkt korrelerte variabler.

Tolkbarhet av resultatene om oppbygningen til en modell avhenger av hvorvidt forklaringsvariablene er informative. Informasjonen i "RawMaterialMix" er et eksempel på lite informativ variabel. Den skal beskrive hvordan blandingsforholdet er i råmateriale ved inngangen av prosessen. Slik informasjon kunne vært nyttig å vite når utfallet skal undersøkes. Imidlertid er det begrenset med oversikt over det faktiske blandingsforholdet, og majoriteten av informasjonen er ukjent. Som vist i Tabell 4.4, er fordelingen mellom kjent og ukjent informasjon meget skjevfordelt. Som beskrevet i Seksjon 3.1.1, er "Unknown mix" ukjent og forskjellige blandinger av de to andre kategoriene "Chicken" og "Turkey". Så en undersøkelse av hvordan kylling og kalkun påvirker resultatet, vil ikke være tilstrekkelig med usikkerhet i blandingsforholdet. Slik var det besluttet å ekskludere "RawMaterialMix" som forklaringsvariabel. Resterende begrunnelser for valg og ekskludering av variabler er lagt til i Vedlegg A.4.1.

Tabell 4.5: Oversikt over korrelasjoner mellom responsvariabler og utvalgte forklaringsvariabler A2 og NIR-målinger.

	A2:	**NIRfat:**	**NIRash:**	**NIRwater:**
Mw:	-0.69	0.33	-0.3	-0.49
SmallMolecules:	-0.26	-0.51	0.49	0.3
BrixAdjusted:	0.41	0.09	-0.07	0.22

4.2.2 Behandling av manglende verdier

Manglende verdier i datasett kan ha en negativ innvirkning på analyse og modellering, fordi flere modeller er avhengige av fullstendig informasjon for å kunne predikere [87]. For å forhindre slike tilfeller, er det nødvendig å behandle de manglende og uidentifiserte verdiene i et datasett. Det kan skyldes blant annet mangler eller feil ved innsamling, tekniske eller systematiske feil ved håndtering av data i lagring eller overføring, eller andre begrensninger som fører til utilgjengelighet [88]. Det finnes flere teknikker for behandling av manglende data, og hver teknikk vil ha en viss form for innflytelse på informasjonen som blir gjenværende etter fullført behandling. Valg av metode må gjøres på en hensiktsmessig måte, slik at datakvaliteten opprettholdet til beste evne [89].

Fjerning av data og fylling av verdier er to metoder for behandling av mangel på data. Valg av metode avhenger av faktorer som typen manglende data og andelen av data som mangler. Fjerning av data innebærer reduksjon i variabler eller observasjoner. Det kan medføre tap av verdifull informasjon og mulig endring i datafordelingen. Imputering av data innebærer å fylle eller erstatte manglende verdier med estimerte verdier [89]. Når disse estimerte verdiene tilføres påvirker det forholdet i datasettet fordi informasjonen opprinnelig ikke er hentet fra selve prosessen. Effekten av metodene skaleres etter andelen manglende data som behandles. Dermed må en være konsekvent over hvilke observasjoner som fjernes eller estimeres, og hvordan effekten kan virke på resultatet.

For visuell inspeksjon av tilstedeværelse av manglende verdier, ble Python-pakken "Missigno" benyttet. Verktøyet bidro til oversikt over omfanget av og distribusjonen av manglende verdier i datasettet. Deretter ble det undersøkt om de manglende verdiene var en del av sekvenser med observasjoner med et fast tidsintervall på 1 minutt. Siden datasettet som tidsseriedata var irregulært, vil mulig metode for imputering måtte kunne ta stilling til det.

LOCF (se Seksjon 2.1.6) ble anvendt for imputering av mangelende data. Ettersom datasettet inkluderte tidsinformasjon og observasjonener fulgte etter hverandre i prosesssen, var det ansett som en gunstig behandlingsmetode. Metoden bruker den siste observerte verdien som et estimat for påfølgende verdier som mangler, til neste observerte verdi. Lengden på sekvensen som det skal imputeres for er dermed avgjørende. Lengre sekvenser kan føre til at estimert verdi blir mindre representativ for den opprinnelige observasjonen. Derfor var det nødvendig å fastsette en grense på antall etterfølgende verdier som skulle fylles. For dette formålet ble ACF-plott for alle variabler benyttet for å identifisere autkorrelasjoner i data. Det muligjorde å finne en grense basert på sterkt korrelasjon. Siden datasettet allerede inneholdt manglende verier og var irregulært, ville ikke ACF-plottet

være presist. Dermed var også første kontinuerlige sekvens av observasjoner på 1 minutt også brukt som data for ACF-plottene.

Som vist i Figur A.2 i Vedlegg A.4.2, var det variabler med NIR-målinger som innehold de største intervalllene med manglende verdier. Terskelen for antall etterfølgende verdier som skulle estimeres for var basert på undersøkler om autokorrelasjon i disse variablene. Korrelasjonskoeffisient på 0.75 var satt som terskel og resulterte i en grense på den femtende observasjonen. 15 minutter var satt på som grense for imputering med LOCF. Sekvenser av manglende verdier over 15 minutter ble fjernet.

Etter behandling ble både observasjoner som var fjernet og beholdt med estimerte verdier undersøkt. Undersøkelsen gikk ut på å identifisere om de behandlede observasjonene var markert eller umarkert. Det var spesielt viktig å få oversikt for om observasjoner med estimerte verdier var markert eller umarkert. Markerte data påvirker begge læringsmetoder, mens umarkerte data kun påvirker semi-veielde metoder.

4.2.3 Encoding av kategoriske verdier

Konvertering av kategoriske verdier til numeriske verdier kalles kategorisk encoding. For å trene og ta i bruk maskinlæringsalgoritmer er denne transformasjonen nødvendig [90]. Det kommer av krav om numerisk informasjon som input for statistiske modeller. Det finnes flere teknikker for å transformere kategoriske verdier. Hvilken teknikk som benyttes avhenger av først og mest av hvilken type kategorisk variabel det er.

De kategoriske variablene i datasettet var av nominell type og har ingen hierarkisk fordeling i kategoriene. For behandling av nominelle kategoriske variabler brukes teknikke kalt "One-Hot Encoding" [17]. Fordelen med teknikken er at egenskapene til nominelle data bevares ved at verdiene forblir ikke-rangerbar. En konsekvens vil være økt dimensjonalitet, når antall variabler øker med antall kategorier i den kategoriske variabelen. Teknikker resluterer til at informasjon av kategoriene representeres i egne variabler. Representasjonen av kategoriene bevares og resultatene av modelloppbygningen er tolkbar.

Den kategoriske variabelen "EnzymeCode" består av enzymtyper, der forholdet ikke kan rangeres på en naturlig måte. Variabelen ble konvertert slik at hver enzymtype var representert som egen forklaringsvariabel. Betydningen av enzymtypene i modelleringsfasen var dermed etter frekvensen av dem i den originale kategoriske variabelen.

4.2.4 Behandling av ekstremverdier

Siste fase av preprosessering av data omhandler behandling av potensielle ekstremverdier og observasjoner. Ekstremverdiene avviker i signifikant grad fra øvrige verdier i datasettet. Observasjonene med slike verdier kan skape vektet "bias" til deres favør. Slike verdier må dermed behandles for å eliminere uønsket "bias" [91]. Eksempler på ekstremverdier kan blant annet skyldes målefeil av instrumenter, menneskelige faktorer eller andre grunner [85]. Det vil være betydningsfullt å håndtere slike verdier for å forbedre kvaliteten på modellering. Ekstremverdier kan bidra til å forstyrre modellens foreståele av data og

tilpasse seg informasjon som avviker seg i høy grad fra øvrig informasjon. En slik innvirkning kan medføre reduksjon i modellens prediksjonsytelse [87].

Datasettet inneholder informasjon fra både standardproduksjon og produksjon under testforsøk. Delsettene skiller seg fra hverandre med hensyn til enzymtypene og råmaterialeblandingene. Det var også gjort endringer i tilførselen av vann og råmateriale, som beskrevet i Seksjon 3.1.3. Som vist i Tabell 4.2, var det stor skjevfordeling i informasjonen som representerte de ulike produksjonene i hele datasettet. Med dette i betraktning ble delsettene inspisert for ekstremverdier separat. For håndtering av ekstremtverdier ble metodene "Local Outlier Factor" (LOF) og "Clusterbased Local Outlier Factor" (CBLOF) fra Python-pakken PyOD (se Vedlegg A.3.4) tatt i bruk.

LOF er designet for å identifisere lokale ekstremverdier (se Avsnitt 2.1.6). Til tross for at datasettet er irregulært som tidsseriedata, inneholder den fremdeles sekvenser av data med faste tidsintervaller. LOF egnet seg derfor som en metode for å identifisere observasjoner som skiller seg fra sine nærmeste naboer. ACF-plottene (vist i Vedlegg av Figur A.3, Figur A.4, Figur A.5) antyder at informasjonen i nærliggende observasjoner vil likne på hverandre. Observasjonene som skiller seg drastisk naboene, vil dermed bli identifisert som ekstrem data. Forklaringsvariabelen "RawMatPercent" har en unik fordeling der var aktuelt å identifisere etter ekstremverdier i lokale områder.

Metoden består av flere parametre for inspisering av ekstrem data, blant annet antall k-naboer. Vurderingen av observasjonene baseres på distansen som er satt fra antall nærmeste nabo. Siden en terskel på 15 ble satt for imputering av manglende verdier (se Avsnitt 4.2.2), ble det aktuelt å definere antall naboer for modellen som det samme.

CBLOF er designet for å identifisere globale ekstremverdier. Modellen ble tilpasset ved å angi antall klynger. Det ble gitt som antall kombinasjoner av de ulike typene enzymer og råmaterialeblandinger.
Modellene angir en score for hver observasjon basert på i hvor stor grad observasjonen regnes som ekstremdata av modellen. Vurdering om ekstremdata var tatt ved hjelp av visuell inspeksjon av score gitt til enhver observasjon. Det var valgt en terskel basert på et skille som er størst mellom utvalgte ekstremdata og resterende datapunkter. Dermed er det lagt til skjønnsmessige vurderinger ved bruk av denne teknikken. Metoden var både benyttet for både normal-sett og design-sett

En variabel som ble tatt spesielt hensyn til under inspeksjon av ekstreme verdier, var variabel "TT12", som viste et sett med temperaturmålinger som adskilte seg fra resten. Fordelingen i "TT12" er vist senere i Seksjon 5.1.

Data som ble vurderte ekstrem i normalsett og designsett var ekskludert fra datasettet før separering. Gjenstående datasett var videre inndelt i markerte og umarkerte data. Hvert delsett ble også inspisert for ekstrem data med LOF og visuell vurdering. Dette var gjort for å forsikre at mulige ekstremdata i alle seksjoner av datasettet var håndtert.

4.3 Modellering

Denne seksjonen presenterer beskrivelsen av rammeverket utviklet for å trene maskinlæringsalgoritmer for å oppnå av målsatte resultater. Dette innebærer begrunnelser for valg av algoritmer og tilhørende beskrivelser. Det gis en detaljert beskrivelse over metoder og teknikker benyttet til oppbygningen av de ulike modellene. Oversikten innebærer blant annet parameterinnstillinger for modellene og optimering, metode for validering og forbedring av modellytelse.

4.3.1 Valg av algoritmer

Følgende delseksjonen forklarer valg av algoritmene i modelleringsfasen. For semi-veiledede modeller, var det valgt algoritmer som representerer de ulike læringene innen semi-veildet læring. Metodene er som følgende: *selv-læring, med-læring og grafbasert-læring*, og er beskrevet i Seksjon 2.1.2. Algoritmene innen semi-veiledet læring er blant annet inspirert av spesifikke veiledede algoritmer. For et rettferdig sammenligningsgrunnlag, blir kun de spesifikke algoritmene fra veiledet læring benyttet i dette arbeidet. Studiet tar for seg undersøkelse av totalt 6 algoritmer, presentert i Tabell 4.6. Videre vises det til nødvendige begrunnelser og beskrivelser av aktuelle algoritmer.

Tabell 4.6: Oversikt over benyttede semi-veiledede og veiledede regresjonsalgoritmer i oppgaven.

Semi-veiledede metoder:	**Semi-veiledede algoritmer:**	**Veiledede algoritmer:**
Selvlæring	Selvtrent-RandomForestRegressor	RandomForestRegressor
Medlæring	CoReg	KNeighbourRegressor
Grafbasert-læring	BHD	SupportVectorRegressor

Selvlæring og "Selvtrent"- RandomForestRegressor

For å undersøke selvlæring i regresjonsproblemer, ble det utviklet en enkel modell basert på prinsippene bak selvlæring og ensemblelæring (Se Seksjon 2.2.1). Modellen, kalt "Selv_RFR", er basert på algoritmen RandomForestRegressor. Modellen betsår av flere beslutningstrær, der den endelige prediksjonen er basert på gjennomsnittet av prediksjoner fra de individuelle beslutningtrærne.

Som nevnt i Seksjon 2.1.2 trenger en algoritme basert på selvlæring en terskel for å avgjøre hvilke datapunkter med pseudomarkeringer som skal inkluderes."Selvtrent RFR" er terskelen basert på modellens "sikkerhet" i prediksjonene. Sikkerheten er basert på evaluering av spredning i prediksjonene på en enkel observasjon fra de individuelle trærne. Stor variasjon i prediksjoner for en observasjon, indikerer stor usikkerhet. Mindre variasjon tyder på høyere sikkerhet. Denne sammenhengen utnyttes for å sette en terskel for inkludering av pseudomarkete data.

Standardavvik er et mål på spredningen i data. For at modellen skulle være uavhengig av størrelsesorden på data den predikerer, var det relative standardavviket benyttet som et bedre estimat.

Pseudomarkerte observasjoner med sikkerhet under avsatt terskel for relativ standardavvik, ble inkludert som en del av treningssettet i neste iterasjon. Denne prosessen fortsetter til maksimal antall iterasjoner er nådd for modelltrening. Prosessen kan også stoppe når det ikke lenger er pseudomarkerte observasjoner som kan kvalifiseres over angitt terskel for inkludering.

Siden Selv_RFR er basert på algoritmen RandomForestRegressor, var RFR også inkludert i forskningen for et rettferdig sammenligningsgrunnlag ved evaluering. Psuedokode 3 presenter pseudokode for algoritmen "Selvtrent"- RandomForestRegressor. $\tilde{X}\_umarkert$ og $\tilde{y}\_umarkert$ beskriver de umarkerte data og tilhørende pseudomarkering, som hadde et relativ standardavvik under gitt terskel ved prediksjoner av de individuelle beslutningstrærne i algoritmen.

Algorithm 3 Selvtrent- RandomForestRegressor

```
1: procedure SelfTrainingRandomForestRegressor                ▷ Konstruktør
2:     maks_iterasjoner ← maksimalt antall iterasjoner
3:     std_terskel ← terskel for relativt standardavvik
4:     model ← RandomForestRegressor(...)                     ▷ Initaliserer modell
5:     X_markert, y_markert, X_umarkert ← None
6: end procedure
7:
8: procedure fit(X_markert, y_markert, X_umarkert)    ▷ Selvtrening av modellen
9:     for i = 1 til maks_iterasjoner do
10:        model.fit(X_markert, y_markert)
11:        ŷ_umarkert = model.fit(X_umarkert)                 ▷ Pseudomarkeringer
12:        for modell.tre in model.trær do
13:            prediksjoner = [modell.tre.predict(X_markert)]
14:        end for
15:        Beregner relativt standardavvik av predikerte verdier
16:        Finner X̃_umarkert                    ▷ X_umarkert under angitt terskel
17:        Oppdaterer X_markert, y_markert ved inkludering av X̃_umarkert og
   ỹ_umarkert
18:        Fjerner X̃_umarkert fra X_umarkert
19:        if ingen X̃_umarkert then
20:            Bryt ut
21:        end if
22:    end for
23: end procedure
24:
25: procedure predict(X)                                      ▷ Predikering
26:    return ŷ = model.predict(X)
27: end procedure
```

Medlæring og COREG

For å vurdere en algoritme som baserer seg semi-veiledet medlæring, ble COREG-algoritmen benyttet fra fra Python-biblioteket "LAMDA" (se Vedlegg A.3.2). Biblioteket har modifisert algoritmen ved å utvide antall definerbare hyperparametere. COREG-algoritmen defineres med hyperparametere som antall maksimale iterasjoner (T) i treningsfasen , distansemetrikker (p_1, p_2) for hver KNR-algoritme og antall nabo (k) felles for begge KNR-algoritmer. Den modifiserte algoritmen valgt å definere k antall nabo for hver KNR-algoritme separat, gjennom (k_1, k_2). Pollstørrelse (P) beskriver størrelsen på det tilfeldig utvalget U' av U for hver iterasjon. I Vedlegg A.7 beskriver Pseudokode 6 versjonen av COREG fra biblioteket.

COREG er basert på den veiledede KNR-algoritmen. Av den grunn ble KNeigborsRegressor-algoritmen (KNR) valgt å undersøkes til i forskningens formål, for å skape et rettferdig sammenligningsgrunnlag for den semi-veiledede algoritmen.

Grafbasert etikettpropagering og BHD - (Domene)

For undersøkelse av semi-veiledet regresjonsmetode som baseres på grafbasert læring, var det tatt inspirasjon fra studiet om "Semi-supervised regression using diffusion on graphs" [74], som forklart i Avsnitt 2.2.2. I studiet ble prediksjonsytelsen til modellen sammenlignet med flere modeller. Støttevektor regresjonsmodellen (SVR) var algoritmen innen veiledet læring som presterte bedre blant modellene sammenliknet med BHD-modellen. Dermed ble SVR benyttet som et godt sammenligningsgrunnlag for modellen.

Til studiet var det lagt ved Python-kode for algoritmen (GitHub:[92]). Koden var imidlertid ikke fullstendig, og lite dokumentert. Av den grunn ble det gjennomført flere endringer og justeringer, slik at algoritmen kunne benyttes til oppgavens formål.

Den modifiserte BHD-modellen består av 4 deler: Først konstrueres det en KNN-graf basert på fullstendig datasett. Det konstrueres en ikke-vektet KNN-graf og RBF-vektet KNN-graf [74]. Deretter beregnes "harmonic score" basert på vektet eller ikke-vektet graf. Følgende initieres "midlertidige markeringer" for umakert data, basert på statistisk informasjon av markerte data. Avslutningsvis benyttes initielle markeringer og tilhørende "harmonic score" til å determinere endelige markeringer for umarkerte data.

Følgende er en oversikt over endringer som var gjort for justering og modifisering av algoritmen i henhold til datasettets størrelse og innhold:

- Algoritmen har 6 parametere:
    1. k: bestemmer antall nabo i en KNN-graf.
    2. $graf\_vektet\_harmonic$: bestemmer om vektet eller ikke-vektet KNN-graf skal benyttes til beregning av "harmonic" score.
    3. $initiell\_respons$: bestemmer statistisk strategi for initiering av midlertidige markeringer for umarkerte data basert på markerte data.
    4. $alpha$: er fra den opprinnelige algoritmen.
    5. "maks_iterasjoner_harmonic" bestemmer maks antall iterasjoner ved beregning av "harmonic score"
    6. "maks_iterasjoner_heat" bestemmer maks antall iterasjoner med påføring av "heat diffusion" på data.
- KNN-graf med eller uten RBF-vekter, fremfor fullstendig RBF-graf.
- I beregning av "harmonic score" kan det velges mellom å bruke vektet eller ikke-vektet graf.
- Antall iterasjoner i beregningen av "harmonic score" er blitt gjort om til parametre, fremfor fast antall iterasjoner på 30.
- Initialisering av "midlertidige markeringer" for umarkerte data kan baseres på domenekunnskap om datasettets innhold. Det er muliggjort for å basere markeringene basert på gjennomsnitt eller median av markerte data etter enzymtype fremfor globalt gjennomsnitt.

Den modifiserte algoritmen tar i bruk domenekunnskap om datasettets innhold for midlertidig markering av umarkerte datapunkter. Dermed en den modifiserte versjonen av BHD-algoritmen, videre omtalt som BHD-Domene. Nedenfor presenteres det pseudokode var den modifiserte algoritmen BHD-Domene:

Algorithm 4 BHD (modifisert) -algoritme

Require: Rå data, Markert treningsdata, Umarkert treningsdata, Testdata, X_trening_markert, X_trening_umarkert, X_test, k, graf_vektet_harmonic, initiell_respons, alpha, maks_iterasjoner_harmonic, maks_iterasjoner_heat

Ensure: y_test_pred, y_trening_pred

procedure BDH ▷ Konstruktør
    k ← k naboer for KNN graf (int)
    graf_vektet_harmonic ← bruk av vektet graf (boolean)
    initiell_respons ← statistsk strategi for initiering av verdier for respons (str)
    alpha ← alpha (float)
    maks_iterasjoner_harmonic ← maks antall iterasjoner for harmonic score (int)
    maks_iterasjoner_heat ← maks antall iterasjoner for heat diffusion (int)
end procedure

procedure FIT_PREDICT ▷ Trening og predikering
    Konstruerer KNN-graf (RBF vektet) med k narboer
    if graf_vektet_harmonic er True **then**:
        Konstruerer KNN-graf (uvektet) med k naboer
        Berenger harmonic score med uvektet KNN graf
    else:
        Beregner harmonic score med RBF vektet KNN graf
    end if
    Initierer utgangspunktsmarkeringer for umarkerte data basert på gitt initieringsstrategi
    Oppdaterer utgangspunktsmarkeringer med harmonic score og heat diffusion
    Predikerer markeringer for testdata og umarkerte data
end procedure

4.3.2 Oppdeling av datasett til unike delsett

Forklare hvorfor datasettet må deles opp i flere delsett.
Modellene kan kun modelleres til en responsvariabel.
For at i semi-veiledet læring er det viktig at de umarkerte datapunktene kommer fra samme fordeling som markerte data.

Responsvariabler

Datasettet består av 3 responsvariabler: Mw, SmallMolecules og BrixAdjusted. Algoritmene som var valgt til studiet hensikt, er begrenset til å modellere med hensyn til én responsvariabel om gangen. Dermed ble datasettet delt opp i tre delsett etter antall responsvariabel. Delsettene bestod av de gjenværende forklaringsvariabelene med tilhørende data og en av responsvariabel hver. Etter oppdelingen til de tre delsettene, ble det mulig å undersøke prediksjonsytelsen til modellene på tvers av datasettene. Selv om informasjonen i forklaringvariablene er lik på tvers av delsettene, vil oppbygningen til modellen være avhengig av informasjonen i responsvariabelen. Informasjonen i hver respons skiller seg fra hverandre, og det er et behov for å modellere hver algoritme etter hver respons. Flere delsett muliggjør ulike typer undersøkelser for prestasjon av veiledede og semi-veiledede metoder på "forskjellige" typer datasett.

4.3.3 Oppdeling av datasett til trening og testing

Som nevnt tidligere i Seksjon 2.1.3, er oppdelingen av trening og testsett essensielt for pålitelig modelltrening og evaluering av ytelse. Strategien som velges for oppdelingen har en avgjørende rolle i evalueringen av prestasjonsytelsen til modellen. Det ideelle målet er å velge en strategi som velger et testsettet kan vurdere generaliserbarheten til modellen, altså evnen til å predikere nøyaktig på usett og uavhengig data.

For regresjonsproblemer benyttes det vanligvis en tilfeldig oppdeling av datasettet. Med kunnskap om at kategoriske variabler har stor betydning for variasjonen i responsvariablene, vil det være suboptimalt med tilfeldig oppdeling. Med datainnsamling av produksjon som er varierende fra dag og uke, er det ikke holdbart å avsette en tilfeldig andel til testing. Det kan føre til at testsettet ikke vil være representativt nok til å vurdere modellens generaliserbarhet.

Et annet viktig aspekt å ta hensyn til er den betydelige autokorrelasjonen som observeres i mange av variablene, som vist av ACF-plottene (Vedlegg). Dermed innehar kontinuiteten i datapunktene betydelig med informasjon for flere av variablene. En tilfeldig utvelgelse av datapunkter til trening og testing, vil potensielt svekke mulig informasjonsgevinst som foreligger i kontinuiteten av data. Samtidig er det stor mangel på kontinuerlige datapunkter, som følge av irregularitet i rå data og behandling av manglende og ekstremverdier.

Formålet med testsettet er å evaluere modellytelse. Dermed må settet bestå av markerte observasjoner som har verdier i responsen. Uten faktiske verdier å sammenligne eventuelle prediksjoner med, mister testsettet sin hensikt. Dermed må testsettet hentes ut fra den

markerte delen av datasettet.

Som et resultat av svaketer ved datasetet som tidsseriedata, samt betydningen av enzymtypene, ble det besluttet å utføre to fremgangsmåter for oppdeling av trening- og testsett.

Uavhengig av tilnærming for oppdelingen, var det valgt en trening og testsplitt på 80/20. Dette var gjort på den markerte andelen av datasettet. For veiledede modeller vil denne fordelingen ivaretas. For semi-veilede modeller vil denne fordelingen imidlertid endres ved inkludering av umarkerte data.

Det er viktig å merke seg at de påfølgende forsøk, kun påvirket inneholdet i avsatt treningdata. Det definerte testsett ble holdt uavhengig og reservert til evaluering av modellytelse.

Alternativ 1: Oppdeling basert på fordeling i enzymtyper

Denne fremgangsmåten tar hensyn til at fordelingen enzymtypene er lik i treningssett og testsett. På den måten sikres det at alle enzymtyper er representert på samme måte i treningsfasen som i testfasen. Strategien blir omtalt som "Alternativ" 1 videre i oppgaven.

Forklaringsvariabelen "råmaterialetype" ble ekskludert grunnet intens skjevfordeling og for å være lite informativ. Dermed ble det besluttet å kun ta hensyn til fordelingen i enzymtypene. Med kun et kriteriet å forholdes til, var det mulig å benytte seg av standardiserte moduler som "RepeatedStratifiedKFold" (fra Avsnitt 2.1.3) fra Scikit-learn biblioteket (Vedlegg A.3.1).

Det er verdt å merke at en slik tilnærming, betraktes som en forenklet strategi for oppdeling av trenings- og testsett for det aktuelle datasettet. Det er ikke tatt hensyn til den daglige og ukentlige variasjonen i data direkte. Mulig variasjon foresaket av forskjellige råmaterialeblandinger er heller ikke tatt i betraktning. Det kan føre til at visse mønstre i datasettet forsvinner, som for eksempel informasjonen som foreligger i kontinuiteten av datapunkter. Imidlertid tillates det for noe tilfeldigheter innad i observasjonene av de forskjellige enzymtyper. Det kan bidra til å redusere risiko for overtilpasning for data.

Alternativ 2: Oppdeling basert på kontinuitet og tidsmessig variasjon

Følgende tilnærming for oppdeling av trening og testdata tar hensyn til flere aspekter som er relevante for modellering. Den sikrer både at den daglige og ukentlige variasjon i datasettet blir adressert, samtidig som den tar hensyn til at kontinuitet i data blir ivaretatt. Hovedfokuset ligger på gjenspeile variasjonen som kommer av den daglige og ukentlige utviklingen. For å oppnå dette, blir de første observasjonene av daglige data for hver uke, gitt en bestemt andel, avsatt til testdata.

Et aspekt som ble tatt hensyn til ved denne tilnærmingen var råmaterialeblandingen til observasjonene."RamMaterialMix" var ekskludert som en forklaringsvariabel fordi informasjon var ekstremt skjevfordelt og lite informativ. Imidlertid er observasjoner knyttet til de forskjellige kategoriene fremdeles en del av datasettet. Majoriteten av de markete observasjonene kommer fra produksjonen under forsøk for ulike råmaterialeblandinger. Når daglige markerte observasjoner inndeles til testdata, kan det hende observasjoner med annen råmaterialeblanding enn "Unknown mix" blir utvalgt. Dersom store andeler

av testdata består av slike observasjoner, kan det hende at treningsdata ikke vil være i stand til å forklare informasjonen i testdata. Det kan forårsake at testsettet er overrepresentert med informasjon som ikke er tilstede i treningssettet. Det kan gå utover kvaliteten på evalueringen av modellen. Dermed var prioritert å inkludere observasjoner som avvek fra ukjent råmaterialeblanding i treningssettet framfor testsett

En annen viktig hensikt med denne tilnærmingen, var å bevare kontinuiteten i data. For å opprettholde sammenhengen i observasjonen, var det forsøkt å avsette en kontinuerlig sekvens med observasjoner til testsettet. Dersom avsatt testandel forårsaker at første sekvens med observasjoner inneholder verdier annet enn "ukjent" for råmaterialeblandingen, gjøres det et forsøk på å finne påfølgende sekvens som fremdeles oppfyller kravet om gitt andel til testdata. Ved mislykket forsøk på å finne en sekvens av opprinnelig ønsket størrelse, forsøkes det å finne lengst mulig seksens som fremdeles tilfredsstiller krav om observasjoner med råmaterialeblanding som er "ukjent".

Nedenfor er det vist til en pseudokode av fremgangsmåten:

Algorithm 5 Trening-testsett-oppdeling med hensyn til dag og uke

```
1: function TRENING_TESTSETT_OPPDELING_DAG(data, test_andel)
2:     trening_markert ← data
3:     test_markert ← liste()
4:     for uke in data_unike_uker do
5:         for dag in data_unike_dag do
6:             data ← [(data_uke == uke) & (data_dag == dag)]
7:             test_rader ← math.floor((data.shape[0] * test_andel))
8:             test_data_dag ← data.head(test_rader)
9:             krav ← test_data_dag[''RawMaterialMix''] =!''Unknown''
10:            if any in krav == True then
11:                while krav.any() do
12:                    test_rader ← test_rader + 1
13:                    test_data_dag ← data.loc[krav.idxmax() :].head(test_rader)
14:                    test_data_dag ← test_data_dag.drop(index = krav.idxmax())
15:                    test_rader ← test_rader − 1
16:                    if (test_rader == 0) or (test_data_dag.shape[0] == 0) then
17:                        pass
18:                    else if (test_data_dag.shape[0] < test_rader) and
   (test_data_dag.index[−1] == data.index[−1]) then
19:                        test_rader ← test_rader − 1
20:                        test_data_dag ← data.head(test_rader)
21:                    end if
22:                    krav ← test_data_dag[''RawMaterialMix''] =!''Unknown''
23:                    test_rader ← test_rader + 1
24:                end while
25:            end if
26:            test_markert ← pd.concat([test_markert, test_data_dag])
27:            trening_markert ← trening_data.drop(test_data_dag.index
28:        end for
29:    end for
30:    treningsdata ← data.drop(test_data)
31:    trening_umarkert ← treningsdata.isna()]
32:    return trening_markert, test_markert, trening_umarkert, treningsdata
33: end function
```

4.3.4 Ulike andeler av treningsdata

Et av hovedformålene med oppgaven var å undersøke i hvilken grad prediksjonsytelsen av semi-veilede modeller skiller seg fra klassiske, veiledede modeller. Videre handlet et av forskningsspørsmålene om å undersøke i hvilken grad ubenyttet informasjon påvirker ytelsen til semi-veiledede modeller, som beskrevet i Seksjon 1.2.1.

For å undersøke disse forskningsspørsmålene, var det forsøkt å studere hvordan inkludering av ulike mengder av treningsdata påvirket modellller av både veilede og semi-veiledede læringsmetoder. I et forsøk på å analysere i hvor stor grad ulike mengder av treningsdata er med på å påvirke ytelsen ble det gjort ulike forsøk. Siden hele datasettet var oppdelt i individuelle datasett for hver responsvariabel, ble det utført ulike forsøk på de ulike datasettene.

Det er viktig å merke at påfølgende forsøk kun påvirket innholdet i avsatt treningsdata. Før forsøkene var oppdeling av Det må merkes at de videre forsøk kun påvirket innholdet i den avsatte treningsdataen. Den definerte testdataen ble satt til side til eventuell evaluering av modell.

Ulike andeler av umarkert treningsdata

For å undersøke hvordan informasjon i umarkerte data påvirker semi-veiledede modeller, ble modellene utsatt for gradvis økning i mengden av inkluderte umarkerte data i treningsfasen. Andelelene var vilkårlige og satt i forhold til størrelsen til det markerte treningssettet: 50 %, 100 % og 200 %. For eksempel, dersom det markerte datasettet består av 100 observasjoner, vil treningssettet til modellen inkluderes med 50, 100 og tilslutt 200 umarkerte observasjoner om gangen. Avslutningsvis ble all tilgjengelig umarkerte data inkludert for å sikre at all data er blitt benyttet.

Undersøkelsen var gjennomført på semi-veiledede modeller i ulike scenarier. Hvert delsett med egne responsvariabler, var fordelt mellom hele datasettet samt designsett. Hvert sett ble deretter oppdelt etter de ulike strategiene for opprettelse av trening og testdata. Undersøkelsen på ulike andeler av umarkerte data var dermed utført ulike kombinasjoner av de overnevnte forholdene. Det ga et bredere grunnlag for å vurdere hvordan inkludering av umarkerte data påvirker ytelse til modellen under forskjellige betingelser.

Strategien for hvordan en bestemt andel av umarkerte data skulle hentes ut av all tilgjengelig umarkerte data, var avhengig av strategien for oppdeling av trening og testdata. Ved valg av ulike strategier, var det mulig å undersøke hvordan innholdet i umarkerte data ville påvirke modellkvaliteten.

For "Alternativ 1: Enzymtyper". var det sørget for at fordelingen av de forskjellige enzymtypene var konsistent mellom treningsett og testssett. Altså ble umarkerte data selektiv hentet ut, slik at det inneholdt samme fordelingen av enzymtyper som var representert i både trening og testdata.

For "Alternativ 2: Dag og kontinuitet" var det fokusert på innehente umarkerte data basert på den daglige og ukentlige variasjonen. I denne sammenheng var ikke fordelingen

i enzymtype vektlagt direkte. De første x umarkerte observasjonene hver dag for hver uke ble inkludert som umarkerte treningsdata. Slik ble også kontinuiteten i data bevart.

Andeler av markert treningsdata

Semi-veilede modeller benytter seg av både umarkerte og markerte data. Læringsmetoden er spesielt nyttig når markerte data er begrenset. For å undersøke hvordan prediksjonsytelsen til semi-veiledede modeller skiller seg fra klassiske, veilede modeller under varierende mengder markerte data, ble det gjennomført ytterligere forsøk med varierende tilgang til markerte data. Formålet var å evaluere hvordan prediksjonsytelsen til modellen i forskjellige scenarier for å utforske hvordan de ulike modelltypene presterer i forhold til hverandre i ulike scenarier.

Forsøkene ble utformet for å evaluere prediksjonsytelse til modell optimert på ulike andeler av tilgjengelig markerte data. Det opprinnelige markerte treningssettet var vilkårlig inndelt i 25 %, 50 %, 75% og 100% av tilgjengelig markerte data. På den måten var det mulig å undersøke hvordan utvikling i prestasjon til modellene med økende datamengder. Spesielt for semi-veilede modeller var det undersøkt for hvordan ulike andeler av umarkerte data påvirket prediksjonsytelsen til modellen basert på de ulike andelene av markerte data.

På grunn av omfanget av forsøkene, som inkluderte både ulike markerte treningsandeler og umarkerte treningsandeler, var de kun gjennomført for en responsvariabel. Tidligere dataundersøkelser viste at enzymtyper påvirket variasjonen i responsene, og dette var mest tydelig i ''Mw". Strategiene for oppdeling av treningsandeler og inkludering av ulike mengder umarkerte data fokuserte på fordeling av enzymtyper. Dermed var det hensiktsmessig å benytte datasettet med ''Mw" som responsvariabel framfor andre, slik at effektene og resultatene av forsøkene ble tydeligere.

For å opprette de mindre treningsandelene av hele treningssettet, ble de ulike strategiene for oppdeling av trening- og testdata anvendt på det opprinnelige treningssettet. Det som ble definert som testdata i denne kontekst, var den andelen av treningsdata som ikke ble gjort tilgjengelig optimering av modellen. For eksempel, hvis 25 % treningsandel ble definert for oppdelingsstrategi ''Alternativ 1: Enzymtype", ble strategien utført på treningssettet, med 75 % av data som ''ubenyttet data". Den ubenyttede delen ble holdt utilgjengelig for modellen under optimering. Imidlertid var hele treningssettet benyttet for evaluering av modellen. På den måten var mulig å undersøke parametere til modellen overtilpasset eller undertilpasset seg med begrenset data.

4.3.5 Med og uten NIR måling

Et av de definerte forskningsspørsmålene var å undersøke i hvilken grad prediksjon av produktkvalitet påvirkes ved bruk spektroskopisk data. For å undersøke dette, ble modellene evaluert basert på datasett både med og uten variabler som inneholdt informasjon fra NIR-sensoren. Undersøkelsen ble gjennomført på alle de separerte datasettene med unike responsvariabler, inkludert hele datasett og designsett, samt de ulike oppdeling strate-

giene for trening og testsett. For denne undersøkelsen ble hele det avsatt treningssettet benyttet til både optimering og evaluering.

I første omgang ble modellene trent på med tilgjengelig forklaringsvariabler, inkludert variabler med informasjon fra NIR-sensoren. Deretter ble de sistnevnte variablene ekskludert fra datasettene. Nye modeller ble deretter trent på de redusere datasettene uten informasjon fra spektroskopi data. Dette gjorde det mulig å evaluere hvordan de ulike modellene presterte på forskjellige mengder og typer informasjon.

4.3.6 Standardisering

Standardisering er en metode for å transformere alle variablene i datasettet til samme skala, som dermed balanserer deres innflytelse under modelleringen. Metoden forbedrer effektiviteten læringen av variablene [87]. Variablene transformeres, slik at gjennomsnittet blir 0 og standardavviket blir 1. Det forsikrer at variablene vektlegges etter betydningen av deres innhold enn størrelse på verdiene.

I modelleringsfasen ble variablene i treningssettet standardisert. De samme skaleringsparameterne ble benyttet for å standardisere testsettet til evaluering. Det blir gjort slik at verdiene i trenings- og testsettet er sammenlignbare [17]. For veiledede modeller gjaldt standardisering kun den markerte andelen av datasettet. For semi-veilede modeller bestod treningssettet både av markerte data og tilført umarkerte data. Treningssettet ble standardisert som en helhet. Skalering av testsettet var dermed basert på skaleringsparameterne fra treningssettet bestående av markerte og umarkerte data.

4.3.7 Kryssvalidering

For å forsikre generaliserbarheten og robustheten til modellene ytterligere, ble kryssvalidering benyttet. Den markerte andelen av treningsdata ble delt opp i trening og valideringsett.

Ved kryssvalidering for semi-veiledede modeller, var treninsettet kombinerte med det aktuelle umarkerte settet som ble tilført i modelleringen. Med andre ord var all tilgjengelig umarkerte data som ble tilført til trening, benyttet under kryssvalideringen med det markerte treningssettet.

Det var nødvendig med ulike kryssvalideringsstrategier utifra hvilken oppdelingstrategi for trening og testsett som var benyttet.

Alternativ 1: Enzymtype For oppdelingsstrategien som sikter på å ivareta fordelingen av en kategorisk variabel i datasettet, var det mulig å implementere seg av kryssvalideringsteknikken RepeatedStratifiedKFold. Teknikken (beskrevet i Seksjon 2.1.3), var benyttet for å sikre en balansert fordeling av enzymtyper i både trening og testdata. Målet var å sette antall segmenter til 10. Stratifisering krever at alle at alle kategorier er fordelt på samme måte i alle segmenter, med minst 1 instans representert i hvert av segmentene. Imidlertid var det tilfeller der kravet ikke ble oppfylt, spesielt for visse undersøkelser som

ble gjennomført. Eksempelvis vil en undersøkelse med 25 % av treningsdata, redusere antall observasjoner av hver enzymtype. I slike tilfeller kan det være begrenset med observasjoner i en kategori til å tillate 10 segmenter.

For å ta hensyn til dette, ble en øvre grense på antall segmenter satt på 10. Dersom treningssettet inneholdt færre enn 10 datapunkter for en type enzym, ble antall segmenter satt til antallet av den enzymtypen som det befant seg færrest av. Det forsikret en systematisk tilnærming til alle gjennomførte undersøkelser, uten nødvendigheten til manuell justering av parameterverdien for antall segmenter.
Antall gjentagelser ble satt til 2. RepeatedstratifiedKfold fra hentet fra Python-biblioteket scikit-learn [93].

Alternativ 2: Dag og kontinuitet For denne oppdelingsstrategien var nødvendig å utvikle en egendefinert metode for oppdeling av trening og validering for å ivareta daglig variasjon og kontinuitet. Antall segmenter var satt til 10 som standard. For å sikre at segmentet for validering inneholdt tilstrekkelig med data, spesielt i tilfeller der det opprinnelige treningssettet er begrenset. Da ble antall segmenter justert med hensyn til laveste antallet observasjoner blant enzymtypene. Tiltaket ble integrert for å minimere forskjellene mellom de ulike oppdelingsstrategiene for trenings- og testsett.

For å sikre at valideringssettene var representative for usett data, ble samme kriteriene som i den opprinnelige oppdelingsstrategien anvendt. Som testsettet, skulle valideringssettet ikke inkludere observasjoner med råmaterialeblandinger annet enn ''Unknown mix". Videre skulle observasjonene i valideringssettet være kontinuerlige og sekvensielle. Pseudokode 7 for den egendefinerte kryssvalideringen er vedlagt i Vedlegg A.7.

4.3.8 Optimering av parametere

For at prediksjonsytelsen til modellene skal forbedres er det nødvendig med justering av parametere til modelllen. Til optimering av modellparamtere har pakken Optuna blitt brukt.

Optuna har en metodisk og effektiv tilnærming til optimering av hyperparametere, ved å tilpasse søke etter optimale hyperparameterkombinasjoner [94]. Metoden er iterativ og er basert på en prøve og feile metode. Prosessen starter med spesifisering for hvilke hyperparametere som skal optimaliseres. Deretter defineres grenser for et ''hyperparameterrom". Rommet skal representere alle mulige kombinasjoner av parametere for den gitte modellen. Optuna utnytter det definerte rommet til å utføre et bestemt antall forsøk med ulike hyperparameterkombinasjoner. Forsøkene gjennomføres der modellen trenes og evalueres med mål om å identifisere de mest optimale parameterkombinasjonene. De beste kombinasjonene velges ut basert på en forhåndsdefinert evalueringsmetrikk.

Hyperparameterområdet tilpasses kontinuerlig basert på tidligere evalueringsresultater. Ved å lære av tidligere forsøk, justeres videre søk til å fokusere på de mest lovende områdene og unngå mislykkede parameterkombinasjoner [94]. Slik tilnærming bidrar til å øke effektiviteten ved å prioritere søkerommet på de kombinasjoner med høyere potensiale til å være mest optimal.

Antallet evalueringsforsøk er definert som parameteren ''trials", og må fastesettes på forhånd. I denne oppgaven var antall ''trials" satt til 100 for alle algoritmer og alle gjennomførte undersøkelser,. Dermed var det mulighet til å evaluere 100 ulike kombinasjoner av parametre for å finne den mest optimale hyperparameterkombinasjonen for hver modell.

En definert evalueringsmetrikk relatert til modellytelse blir brukt av Optuna til å vurdere og sammenligne forskjellige kombinasjoner av hyperparametere. I denne oppgaven var gjennomsnittlig RMSE-score på valideringssettene i en kryssvalidering valgt som grunnlag å basere vurderingen på. Kryssvalidering ble benyttet for å sikre at valg av beste hyperparamterkombinasjon for modellen var basert på pålitelige og robuste vurderinger.

Optimering av COREG-modell

Følgende viser til avsatt søkerom for hyperparametere som var definert for optimering av COREG. For k antall naboer og p-distansemetrikk var det valgt intervall med tallverdier, som vist i Tabell A.7 i Vedlegg. For antall maksimale iterasjoner (T) og pollstørrelse (P) var det å definere faste verdier innad i fasen under optimalisering av modellen og andre verdier for den modellen som var blitt utvalgt etter fasen.

Maksimale iterasjoner (T) var definert utifra antall umarkert data og tid beregnet for optimering og evaluering. I utgangspunktet er (T) satt til halvparten av antall umarkerte data. Ved tilfeller med 1000 umarkerte data, var maks antall iterasjoner satt til 500. Deretter var parameteren justert etter tid avsatt til optimalisering av en modell og evaluering av endelige og optimale modell. Parameterverdien var nedjustert etter antall operasjoner som var nødvendig å gjennomføre i fasen. Under optimaliseringsfasen var antall operasjoner en kombinasjon av antall ''trials" for Optuna, antall segmenter og muligens repetisjoner i kryssvalidering, samt maksimal antall iterasjoner i COREG-modellen. Hver operasjon var antatt å utføres på 1 sekund. Maksimale iterasjoner (T) var nedjustert som en beregning av tiden for alle operasjoner å gjennomføre. Pollstørrelse (P) var definert som en funksjon av (T), med en øvrig grense på 250.

I Vedlegg inneholder Pseudokode 8 en beskrivelse for valg og beregning av parameterverdier for (T) og (P) som en funksjon av tid. Under optimaliseringsfasen var det valgt å sette av en øvrig grense på 6 timer. For evaluering av endelig modell som var utvalgt etter optimering, var det satt en øvrig grense på 12 timer.

Optimering av BHD-modell

Tabell 4.7 viser til avsatt søkerom for hyperparametere som var definert for optimering BHD-algoritmen. Det skal merkes at parameteren k som definerer antall naboer i "KNN"-grafen som konstrueres, fikk en øvrig grense. Grensen var satt var en konsekvens av begreningser i RAM, som beskrevet i Avsnitt 1.3. Øvre grense var satt til maksimal antall datapunkter som maksimerte tilgjengelig minneplass ved konstruering av graf. Nedre grense var satt relativt i forhold til dem maksimale grensen. På den måten er det mulig å unngå trening svake modeller under optimeringen fordi lavere antall naboer er

ofte forbundet med svakere læring av informasjonen i datasettet.

Muligheten for å velge mellom vektet og ikke-vektet KNN-graf ved beregning av $harmonic_score$ ble lagt til. Modellen benytter seg av statiske resultater basert på kategorisk fordeling i gitt datasett, av den grunn ble det gitt et valg om bruk av gjennomsnitt eller median.

Tabell 4.7: Parametergrid for BHD.

Hyperparameter:	**Type:**	**Søkeområde:**
"k"	Heltall	["k_min", "k_maks"]
"graf_vektet_harmonic"	Kategorisk	[False, True]
"initiell_respons"	Kategorisk	['domain_mean', 'domain_median']
"alpha"	Flyttall	[0.0, 1.0, (step = 0.001)]
"maks_iterasjoner_harmonic"	Heltall	[1, 100]
"maks_iterasjoner_heat"	Heltall	[1, 100]

4.4 Evaluering

Evaluering er en sentral fase i CRISP-DM metodikken, hvor de utformede og optimaliserte modellene gjennomgår en grundig vurdering. Trinnet er betydelig for å vurdere modellkvalitet og kvantifisere modellens nøyaktighet og prediksjonsevne. Det bidrar til å identifisere både styrker og svakheter ved modellene.

Det er utført både en kvantitativ og kvalitativ vurderinger. Den kvantitative vurderingen innebærer benyttelse av ulike evalueringsmetrikker innen regresjon. Det gir en teoretisk og objektiv vurdering av modellens ytelse. Den kvalitative vurderingen involverer bruk av en forenklet flermålsanalyse (MCDA). Analysen muliggjør sammenligning av modellene i praktiske sammenhenger basert på flere kriterier. Slik vurdering er relevant for beslutningstaking ved reell integrasjon av modellene i virkelige prosesser. Et bredt evalueringsgrunnlag på denne måten, gir mulighet for omfattende sammenligning av modellene og fremhever i hvilken grad de kan bidra til å løse den definerte problemstillingen.

4.4.1 Evalueringsmetrikker

For å forstå og vurdere nøyaktigheten til maskinlæringsmodellene, ble de anvendt og evaluert på et dedikert testsett. Testsettet ble adskilt fra treningssettet før modelltrening og optimering var igangsatt. I løpet av treningsfasen ble valideringssett konstruert gjennom kryssvalideringsteknikker, for objektiv evaluering modellens evne til å predikere nye og usette data.

Modellene ble evaluert på flere ulike evalueringsmetrikker. Hver metrikk beskriver og informerer om spesifikke aspekter ved modellens ytelse. Sammen bidrar metrikkene til en helhetlig vurdering av styrker og svakheter ved modellen. Innsikt i flere aspekter gir også verdifull informasjon om mulige forbedringer. En slik totalvurdering danner et solid

sammenligningsgrunnlag for ulike modeller.

Evalueringen av modellene ble utført ved følgende metrikker: RMSE, R^2, MAE og MAPE (forklart i Seksjon 2.1.4). I treningsprosessen var det lagt spesiell oppmerksomhet på RMSE og R^2-scorene. Målet under trening og optimering av hyperparametere til modellen, var å minimere RMSE-scoren ved anvendelse på valideringsett. Risikoen for over- og undertilpasning ble vurdert gjennom sammenligning av R^2-scoren på trening, validering og testsettet.

Både RMSE og MAE gir informasjon om avvik mellom predikerte verdier og faktiske verdier av datapunkter. Som nevnt i Seksjon 2.1.4, har RMSE en større følsomhet for ekstreme verdier, sammenlignet med MAE. Siden MAE vekter alle feil likt og er dermed mindre påvirket. For å fremme modellens følsomhet overfor ekstreme verdier, ble en evaluering basert på den relative differansen mellom metrikkene gjennomført. Et betydelige avvik vil indikere at modellen er mer følsom for ekstreme verdier. Den relative differansen var beregnet som følger:

$$\text{RMSE/MAE (\%)} = \frac{\text{RMSE} - \text{MAE}}{\text{MAE}} \tag{4.1}$$

4.4.2 MCDA-Analyse

Denne seksjonen vil foreta seg en beskrivelse av MCDA-anelysen som blir gjennomført. Først introduseres relevante begrensninger og antagelser som ble satt for å kunne gjennomføre en slik analyse. Deretter forklares verktøyet som ble benyttet for å gi et bredere beslutningsgrunnlag på evaluering av anvendelsen for ulike varianter av modeller og data.

Begrensninger og antagelser

For å kunne gjennomføre en forenklet versjon av MCDA-analysen har det vært nødvendig å ta i betraktning til flere begrensninger og gjøre visse antagelser. På den måten ar det mulig å utføre en vurdering på de ulike de ulike alternativene ved mulige implementering av de forskjellige kombinasjonene av algoritmer av læringsmetodene og tilgangen til spektroskopisk data. Begrensingene og antagelsene påvirker omfanget og nøyaktigheten av analysen, og må tas hensynt il ved endelige vurderinger.

En betydelig begrensninger har vært begrenset tilgang og mangelen på tilstrekkelig kvantitativ data. Dette har ført til at analysen må baseres på antagelser og strategiske vurderinger enn faktiske tall og verdier. Faktiske verdier knyttet til produkt, sensor og utstyrskostnader og andre økonomsike variabler har vært utilgjengelig på grunn av sensitivitet eller begrensinger i anskaffelsen av informasjonen.

På grunn av begrensningene har vurderinger av økonomsike aspekter og andre relevante faktorer blitt gjennomført på et mer strategisk og overordnet nivå. Dette har en påvirkning nøyaktigheten og grad av detaljer i evaluringene av alternativene for mulig prosessstyring og forbedring. Det er dermed ikke tatt høyde for andre mulige tiltak

som eventuelt gjøres for å realisere de muligheter som blir presentert ved hvert alternativ.

Det er heller ikke gjennomført en behovsanalyse som står for sentral en flermålsnalayse. En slik analyse bidrar til å identifisere problemet som skal løses og at for å forsikre at alternativene i MCDA-analysen er hensiktsmessige [76]. En dekkende behovsanalyse hadde hadde nyttig for å forsikre at analysen tar viktige og faktorer i betrakning med hensyn til relevante interesser og preferanser. Sensitiv og begrenset informasjon om prosessen og produksjonen har vært årsken til at en behovanalyse ikke ble gjennomført.

Disse begrensningene og antagelsene må tas i betraktningen når evalueringene og resultatene av analysen tolkes og benyttes som et beslutningsgrunnlag valg av alternativer. Begrensningene spiller en avgjørende faktor på analysens nøyaktighet, pålitelighet og validitet. Analysens vurderingsgrunnlag kan dermed være utsatt for endringer ved tilgjengeliggjøring av informasjon.

Mulighetsstudie

Mulighetsstudiet tar for seg undersøkelse av tre ulike alternativer i tilegg til dagens prosess, for muligens å forbedre prosessen i produksjonen. Formålet med analysen er å vurdere alternativene på et bredere grunnlag og undersøke flere aspekter av alternativene ved eventuell implementering. Vurdering baseres seg hovedsakelig på et teknisk grunnlag og inkludering av økonomisk askpekt.

Dagens prosess

Den nåværende prosessen er beskrevet i Avsnitt 3.1.1. Dagens praksis innebærer at produktkvalitet ikke måles før i ettertid i produksjonen, og det gjøres ingen beslutninger basert på grad av kvalitet på produktet underveis. Det innebærer at det ikke brukes maskinlæringsalgorimter til støtte og vurderinger. Prosessen foregår med standardverdier for styringsparametre, og det gjøres ingen justeringer underveis basert på variasjon i innholdet i råmateriale. Det kommer av at det ikke er utplassert en sensor for målingen av spektroskopisk data om materialet. NIR-sensoren var kun midlertidig utplassert i forsøksperiodene, og er ikke del av standardprosessen.

Dagens situasjon er ansett som et nullalternativ. Det setter grunnlaget analysen, ved at det kan vurderes om det bør gjøres endringer eller ikke. Om dagens prosess kan evalueres til den grad iforhold de foreslåtte alternativer, at å unngå implementering av andre løsninger. Eller at de foreslåtte alternativer fremmer gode nok resultater til at det pålitelig og valid erstatning for dagens løsninger.

Alternativ 1 - Prosessstyring med veiledede maskinlæringsalgoritmer og NIR-måler

Alternativ 1 innebærer anvendelse av klassiske og veiledede maskinlæringsalgoritmer for å utvikle en soft-sensor som kan predikere produktkvaliteten under produksjonsprosessen. Disse algoritmene krever et fullstendig datasett med referanser på kvalitetsmålinger for

alle data. Implementering av alternativet forutsetter tilgang til spektroskopisk data fra en NIR-sensor. Informasjon fra tradisjonelle måleinstrumenter, samt spektroskopisk informasjon om råmaterialet, vil muliggjøre aktive justeringer av styringsparametere underveis i prosessen for å muligens optimere produktkvaliteten. Tidligere målinger på produktkvalitet i prosessen kan også bidra til effektivisere beslutninger tatt produkter basert på kvalitet.

Alternativ 2 - Prosesstyring med semi-veiledede maskinlæringsalgoritmer og NIR-måler

Alternativ 2 innebærer også anvendelse av maskinlæringsalgoritmer for å utvikle en soft-sensor, men med semi-veiledede algoritmer. Disse algoritmene krever ikke et fullstendig datasett. Implementering av dette alternativet forutsetter også tilgang til spektroskopisk data fra en NIR-sensor. Dette alternativet er viktig, siden det kan ha redusert krav på innsamling av kvalitetsmålinger. Modellen kan også ta i bruk kontinuerlig innsamlet data underveis i produksjonen, uten referanser i kvalitetsmålinger.

Alternativ 3 - Prosesstyring med maskinlæringsalgoritmer uten NIR-måler

Alternativ 3 innebærer anvendelse av maskinlæringsalgoritmer for å utvikle en soft-sensor, uten tilgang til spektroskopisk data. Modellen er begrenset til tilgang på informasjon fra tradisjonelle måleinstrumenter til å predikere produktkvalitet. Alternativet dekker både veiledede og semi-veiledede algoritmer.

Evalueringskriterier

Det vil bli presentert ulike kriterier for å evaluere alternativene. Hvert kriteriet har en skala fra $1-5$, der 5 representerer høyest score og 1 lavest score. Alle alternativene blir vurdert etter hvor godt de tilfredsstiller de respektive kriteriene. Følgende kriterier består av; "Prediksjonsytelse og Modellkvalitet", "Funksjonalitet og kompleksitet", "Verdiskapning", og til slutt, "Investering og vedlikehold".

Tabell 4.8: Oversikt over evalueringskriterier og poeng-score knyttet til krav av tilfredsstillelse.

Score:	Prediksjonsytelse og Modellkvalitet:	Funksjonalitet og kompleksitet:	Verdiskapning:	Investering og vedlikehold:
1	Svært lav nøyaktighet og modellkvalitet	Svært lang modelleringstid og høye krav til minneplass	Minimal eller ingen verdiskapning	Svært høy kostnad og krevende vedlikehold
2	Lav nøyaktighet og modellkvalitet	Lang modelleringstid og betydelige krav til minneplass	Begrenset verdiskapning	Høy kostnad og krevende vedlikehold
3	Moderat nøyaktighet og modellkvalitet	Moderat modelleringstid og moderate krav til minneplass	Moderat verdiskapning	Moderat kostnad og vedlikeholdskrav
4	Høy nøyaktighet og modellkvalitet	Kort modelleringstid og lave krav til minneplass	Høy verdiskapning	Lav kostnad og vedlikeholdskrav
5	Svært høy nøyaktighet og modellkvalitet	Svært kort modelleringstid og minimale krav til minneplass	Maksimal verdiskapning	Svært lav kostnad og vedlikeholdskrav

Prediksjonsytelse og modellkvalitet
Dette kriteriet vurderer nøyaktigheten og feilrate til modellens prediksjoner. Evalueringen baseres på teoretiske evalueringsmetrikker som RMSE og R^2. Lavere RMSE-score indikerer bedre nøyaktighet, mens høyere R^2-score indikerer større pålitelighet i modellens prediksjoner.

Modellkvalitet omfatter modellens robusthet og evne til å opprettholde nøyaktige prediksjoner under påvirkning av støy eller ekstreme data [77]. Denne vurderingen inkluderer analyse av variasjonen i resultater på tvers av ulike evalueringsmetrikker, samt sammenligning av modellens ytelse i trenings-, validerings og testdata. Det er også viktig å vurdere modellens konsistens på tvers av datasett med forskjellige kvalitetsmålinger for å sikre robusthet.

Funksjonalitet og kompleksitet
Kriteriet om funksjonalitet omfatter vurdering av modellens effektivitet i prediksjoner, gjennom modelleringstid og ressursforbruk. Dette måles blant annet ved tiden brukt på optimalisering og trening av modellen, samt minneplass som kreves under prosessen. Lavere kompleksitet og modelleringstid er ønskelig for å sikre effektiv implementering og skalerbarhet [77].

Verdiskapning
Følgende kriteriet tar utgangpunkt i å evaluere alternativets muligheter til å bidra til verdiskapningen i produksjonsprosessen. I fravær av kvantitativ data, vurderes potensialet for verdiskapning basert på modellens mulige anvendelse. Dette innebærer ressursutnyttelse og evnen til å øke produksjonsverdien gjennom beslutninger basert på modellens prediksjoner på produktkvalitet.

Investering og vedlikehold
Dette kriteriet vurderer de øvrige økonomiske krav ved implementering og drift av modellen over tid. Det inkluderer eventuelle investeringer av sensorer for modellfunksjonalitet, samt vedlikeholds og operasjonelle kostnader. Videre dekker det også kostnader knyttet til krav til datainnsamling og andre ressursbehov.

Vektingen av kriteriene

Prediksjonsytelse og modellkvalitet
Dette kriteriet er vektet høyest av kriteriene på 30 %. Det er vektet høyt fordi pålitelige og nøyaktige prediksjoner er avgjørende for implementering av maskinlæring i prosesssen. Modellens evne til å opprettholde nøyaktighet på tvers av kvalitetsmålinger og data med støy og ekstremverdier er kritisk for å sikre robusthet og pålitelighet i produksjonsmiljøet. Høy nøyaktighet og lav feilrate validerer modellen og øker tilliten til at modellen av kan gi pålitelige resultater til utforming av beslutningsgrunnlaget.

Funksjonalitet og kompleksitet
I produksjonssammenheng er det viktig at modellene er effektive i modelleringstid og ressursforbruk. Det er essensielt for praktisk implementering og skalerbarhet. Modellen vil være foretrukket og gjennomførbar i en produksjonssetting, dersom modellen er rask og mindre ressurskrevende. Dermed det vektet med 25 %.

Verdiskapning
For å rettferdiggjøre investeringen i modeller og maskinlæringsteknologi, er det viktig at det kan bidra til verdiskapning. Dette innebærer utnyttelse av ressurser, forbedringer i beslutningsstøtte og muligheten til å øke produksjonsverdien gjennom tidligere predikereringer og vurderinger på på produktkvalitet. Imidlertid vil ikke vurderinger være holdbare uten kvalitative data, og kriteriet vektes dermed lavest på 20 %.

Investering og vedlikehold
Dette kritereiet er knyttet til implementering og vedlikehold av modellen. Det inkluderer investeringer i nødvendig teknologi, som NIR-sensor, og gjennomgående operasjonelle kostnader. Kostnadsbildet er et viktig aspekt å ta stilling til vurdering om implementering av modellen. Siden investeringen og kostnadene legger grunnlaget for hvordan modeller av ulike forutsetninger opererer. Dette kriteriet er heller ikke like holdbart uten kvalitativ data, og kriteriet vektes lavere på 25 %.

Tabell 4.9 oppsummerer vektingen av de ulike kriteriene for MCDA-analysen.

Tabell 4.9: Vekting av kriterier for MCDA-analyse.

Kriterier:	**Vekt:**
Prediksjonsytelse og modellkvalitet	30 %
Funksjonalitet og kompleksitet	25 %
Verdiskapning	20 %
Investering og vedlikehold	25 %

Beregning av totalscore

Hvert alternativ vil blir vurdert etter de definerte kriterier og tilhørende grunnlag. Hver score innen hvert kriteriet vil deretter vektes i henhold til kriteriets forhåndbestemte vekt. De vektede scorene vil tilslutt summeres og gi en totalscore for hvert alternativ. Det alternatvet med høyest score vil bli ansett som det foretrukkede alternativet basert på MCDA-analysen som har blitt gjennomført.

Beregning av totalscore gjennomføres for hvert alternativ basert på Formel 4.2. Totalscore er en vektet sum av produktet mellom score for hvert kriterium. $score_i$ beskriver score gitt for kriteriet nr. i og $vekting_i$ beskriver tilhørende vekt for kriteriet nr. i.

$$Totalscore = \sum_{i=1}^{n} (\text{score}_i \times \text{vekting}_i) \tag{4.2}$$

4.4.3 Benchmarking

Ved evaluering utføres det beregninger på evalueringstrikker som blant annet RMSE. Slik metrikker mål på avvik relativ til målevariabelen som vurderes. Imidlertid er det krevende å vurdere hvilken grad av feilmargin som kan anses som akseptabel margin.

Ved fravær av en terskel for hva som kan anses som akseptabel feilmarginer for prediksjoner utført av maskinlæringsalgoritmene, ble globale statistiske bereninger utført på datasettet. Siden de kategoriske variablene utgjør betydelig andel av produksjonen, settes terskler for feilmarginer i målverdi basert på dem. Dermed ble det valgt å bruke de gjennomsnittlig verdier av målverdiene basert på enzymtyper og råmateriale som benchmarks for maksinlæringsalgoritmene som blir evaluert. Disse vises henholdvis i Tabell 5.1 og Tabell 5.9 for hver testdata.

4.5 Bruk av kunstig intelligens (KI)

I denne oppgaven har kunstig intelligens (KI) blitt brukt som et hjelpemiddel. KI har blitt brukt som et verktøy i forskningsperioden, der hensikten var å effektivisere repetitive prosesser og finne eventuelle feil.

KI ble brukt for rettskrivning, der fokuset var å få tilbakemeldinger på setningsoppbygningen og andre grammatiske feil. Dette var en metode som effektiviserte prosessen, i tillegg til å få muligheten til å lære av gjentagende feil. Det ble rettet på konkrete feil som førte til at behovet for å lete etter samme feil på andre nettsteder minket. Istedenfor å kun bruke en ordbok til rettskrivningen, ble mulighetene med KI utforsket i tillegg.

Underveis ble det oppdaget at KI ikke var like relevant og hjelpsom for mange av utfordringene som ble møtt. KI ble for eksempel brukt for å oversette utklipp av en tekst. Da ble det funnet en feil i oversettelsen. Dette er en av flere eksempler på at KI ikke er feilfri. Dermed understrekes det at man må være klar over at verktøyet kan ta feil og at det må brukes med måte, akkurat som alle andre hjelpemidler.

En liste over KI modeller som ble brukt:

- Sikt KI-chat
- Gemini AI

Kapittel 5

Resultater

Følgende kapittel innebærer presentasjon av funn fra analyser gjennomført i tråd med utvalgt metodologi. Kapittelet er inndelt i flere seksjoner. Det vises til sentrale observasjoner oppdaget av undersøkelser av datasettet, som var avgjørende for hvordan endelig funn var anskaffet. Videre presenteres resultater om vurdering og ytelse av modeller. Resultatene er inndelt etter type kvalitetsmåling, datasett og alternative fordelinger av data til testing. Avslutningsvis vil kapittelet bestå av resultater av MCDA-analyse utført. Funnene fra deler av CRISP-DM metodikken, med en beredere evaluering fra MCDA-analysen, vil gi en helhetlig vurdering på observerte mønstre og sammenhenger til studiets formål.

5.1 Forundersøkelser og behandling av datasett

Følgende seksjon vil vise til sentrale observasjoner funnet gjennom undersøkelser av datasettets innhold, som beskrevet i delkapittel 4.1. Det er valgt å fremme forståelse av datasettet, siden egenskapene skilte seg vesentlig fra forventningene om data fra en industriell produksjon. Egenskapene av innholdet til datasettet, var grunnlaget for flere metodiske valg gjennomført i studiets gjennomførelse. Generelt vil innholdet avgjøre hvilke behandlingsmetoder som må gjennomføres. I denne sammenheng var visse særtrekk i innholdet, som stod til grunn for utformingen av flere ved undersøkelsene til forskningen. For å illustrere faktorene som lå til grunn for flere metodiske valg, vil det vises fram til blant annet visualiseringer av datasettets innhold før behandling.

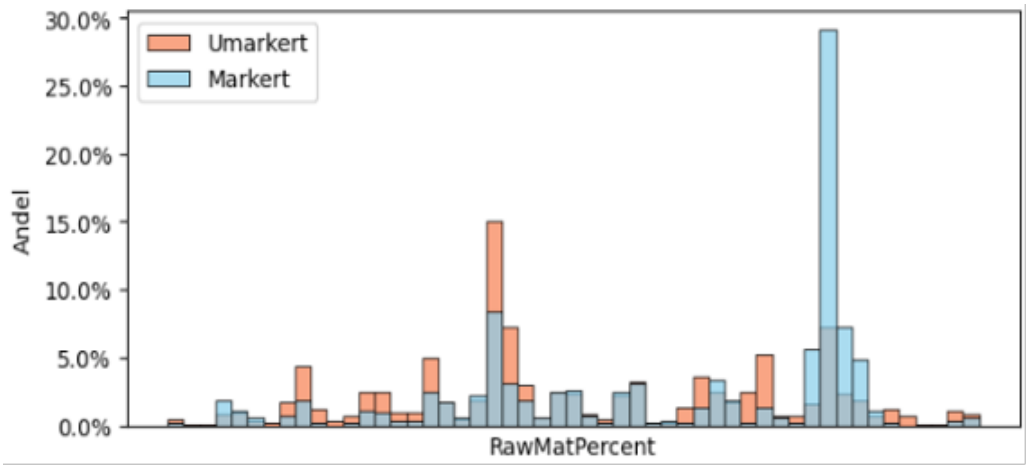
Visualiseringer

Histogrammer er benyttet for å illustrere ulike fordelinger av data i bestemte variabler. Datasettet inneholder ulike typer variabler som kan inndeles i to hovedkategorier: kontrollerbare styringsparametere som temperatur og strømning, og ukontrollerbare målinger som næringsinnholdet i råmaterialet. I tillegg består datasettet av flere undergrupper og kategorier. Markerte og umarkerte observasjoner, samt observasjoner knyttet til spesifikke kategorier som enzymtyper. Disse undergruppene er tatt i betraktning ved utformingen av histogrammene for de ulike variablene av interesse.

Det må understrekes at datasettets innhold er delt mellom markerte og umarkerte data. Histogrammene viser fordeling av disse to typene data separat for hver variabel, i samme graf. Søylehøyden representerer dermed ikke andelen av observasjoner i hele datasettet. Søylehøyden beskriver heller andelen av verdienene i de markerte og umarkerte data separat. Siden majoriteten av datasettets innhold består av umarkerte data, ville ikke andeler av markerte data av hele datasettet vært synlig i grafen. Fordelingen hadde vært vanskelig å antyde.

Figur 5.1 illustrerer fordelingen av forklaringsvariabel "RawMatPercent" i datasettet, inndelt i markerte (blå) og umarkerte (oransje) data. Histogrammet viser til fordeling med flere topper. En slik fordeling, omtalt multimodal, indikerer at målingene grupperer seg rundt spesifikke verdier. Dette kommer av at måleinstrumentet beskriver forskjellige underliggende prosesser. Toppene reflekterer ulike innstilliger satt for de kontrollerbare styringsparametere som strømning av råmateriale og vannstrømning i prosessen. De mindre toppene omkring de større toppene representerer de mindre variasjonene rundt disse innstillingene.

Fordelingen i grafen er relativ til den bestemte delen av datasettet. Høydene på søylene er dermed ikke direkte sammenlignbare. Det er formen på fordelingen og plassering av toppene som er av interesse. Både markerte og umarkerte data viser en multimodal fordeling som er lignende, med topper på relativ samme steder på grafen. Imidlertid viser markerte data en mer skjev fordeling med den store blå søylen til høyre. Dette indikerer at en større andel av markerte data består av høyere verdier for variabelen enn i de umarkerte data.

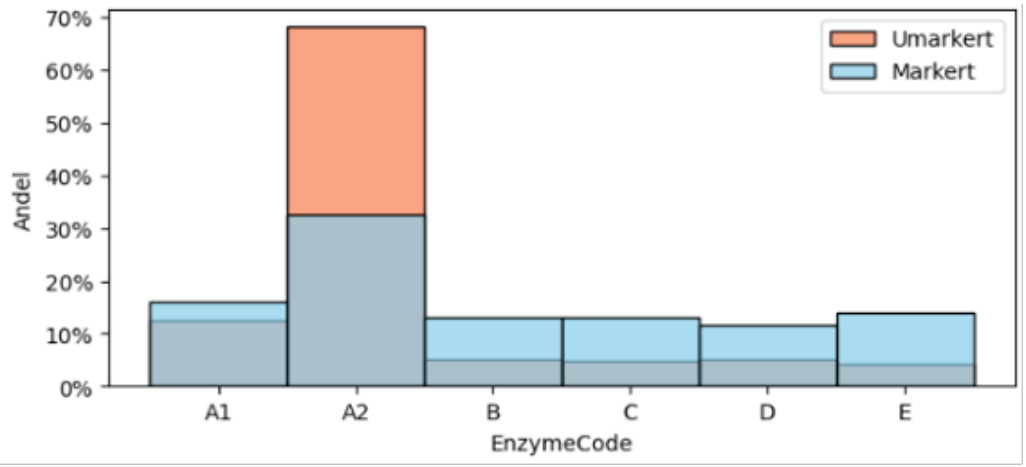


Figur 5.1: Histogram over "RawMatPercent". Markerte datapunkter (lys blå) og umarkerte datapunkter (oransje).

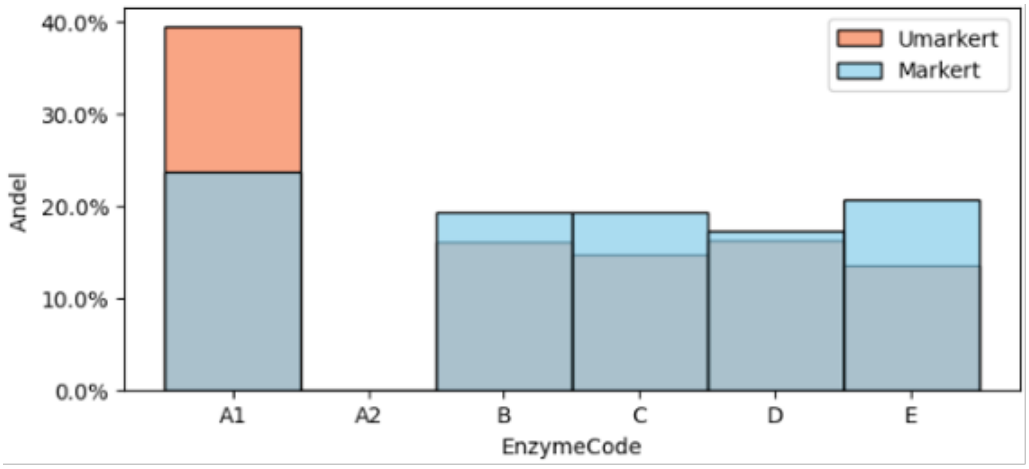
Figurene nedenfor illustrerer fordelinger av enzymtypene i markerte data og umarkerte data. Fordelingen er av spesiell interesse, siden enzymtypene er forbundet med hva slags produksjonstype de beskriver, som forklart i Seksjon 3.1.3.

Figur 5.2 viser at majoriteten av de umarkerte datapunktene er knyttet til enzymtype "A2", etterfulgt av "A1". Det skyldes av at produkter av "A2", som er standardenzymet i normalproduksjon, i store deler ikke var systematisk merket utenfor testforsøkenes perioder. Til kontrast er fordelingen av markerte data jevnere på tvers av enzymtypene. Dette kommer av den systematiske innsamling av laboratoriemålinger under forsøksperioden med ulike enzymtyper. En slik skjevhet mellom markerte og umarkerte data kan føre til at spesifikk informasjon blir overrepresentert når de benyttes.

Figur 5.3 viser fordelingen av data etter enzymtyper ved ekskludering av standard enzymtype "A2". Innholdet i figuren viser fordelingen av data av de resterende enzymtypene som representerer designproduksjonen. Fordelingen i markerte og umarkerte er mer balansert i dette tilfellet. Imidlertid er det fremdeles overvekt av umarkerte data knyttet med et av enzymtypene, "A1". Selv i informasjonen fra designproduksjon er det en skjevhet, som gjør at informasjonen mellom markerte data og umarkerte data ikke reflekterer hverandre.



Figur 5.2: Figuren viser til et histogram over fordeling av enzymtypene i hele datasettet. Innholdet er fordelt i markert (lys blå) og umarkert (oransje) observasjoner. x-aksen beskriver de aktuelle enzymtypene representert i hele datasettet. y-aksen beskriver andelen av totale markerte eller umarkerte observasjoner som er representert i de respektive enzymtypene. Figuren viser at enzymtype "A2" har dominerende andel av informasjon i datasettet, spesielt i den umarkerte andelen.



Figur 5.3: Histogram av fordeling av enzymtypene i markert del (lys blå) og umarkert del (oransje) i data fra designproduksjon, uten enzymtype A2.

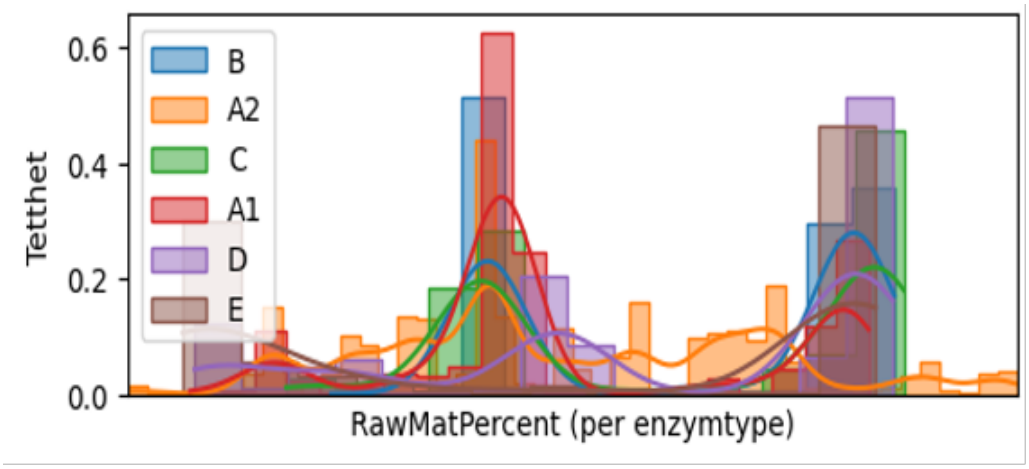
Figurene som følger, viser til hvordan valg av enzymtype som er forbundet med type produksjon, har spilt en rolle for prosessparametere og produktkvalitet. For å tydeliggjøre sammenligning er det valgt å legge til kjernedestimeringsmetode (KDE), som gir mer glatte kurver over histogrammene.

Figur 5.4 viser til fordelingen av forklaringsvariabelen "RawMatPercent", der informasjonen er kategorisert etter variabelen "EnzymCode". Figuren viser at målinger knyttet til standardproduksjon og enzymtype "A2" (oransje), har en distinkt fordeling sammenlignet med de resterende enzymtyper og designproduksjon. Fordelingen for målinger knyttet

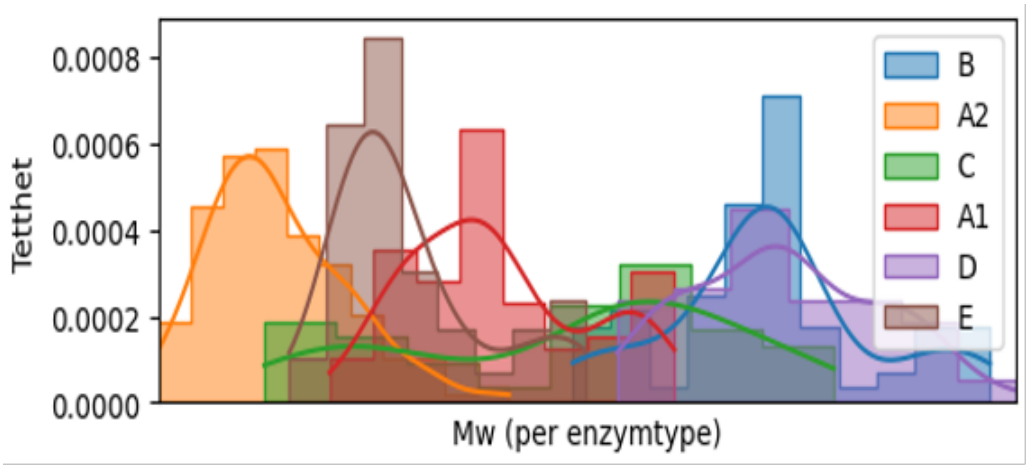
"A2" er jevnere og har ikke den markante toppen til høyre i grafen som er til stede for andre enzymtyper. Basert på det, kan det antydes at strømningen av vann og råmateriale under standard produksjon er mer stabil og balansert enn under designproduksjon. Dette gir en indikasjon på hvordan informasjonen fra standardproduksjon og designproduksjon skiller seg fra hverandre på flere måter enn kun valg av enzymtype.

Figuren, sett i sammenheng med Figur 5.1, viser hvordan informasjonen av markert og umarkert data henger sammen med enzym- og produksjonstype. Det er markant topp til høyre i begge figurerer, som representerer hvordan informasjon i de markerte data kommer hovedsakelig fra designproduksjonen. Det gir videre indikasjoner på hvordan informasjonen i markerte og umarkerte data skiller seg fra hverandre.

Videre vises det til en graf som demonstrerer hvordan valg av de forskjellige enzymtypene gjenspeiles i kvalitetsmålingene. Figur 5.5 illustrerer fordelingen i responsvariabel "Mw", er kategorisert etter enzymtype. Grafen viser at hver enzymtype gir oppgaven til en unik fordeling av verdier for responsen. Enzymtype "A2" (oransje) viser til gjennomgående lavere verdier enn de resterende enzymtyper. Dette typer på at produkter i standardproduksjon kan gi lavere verdier på kvalitetsmålingen, sammenlignet med enzymtypene som det var blitt gjort forsøk med i forsøksperioden.



Figur 5.4: Histogram over fordelingen av variabel "RawMatPercent" etter tilhørende enzymtype.



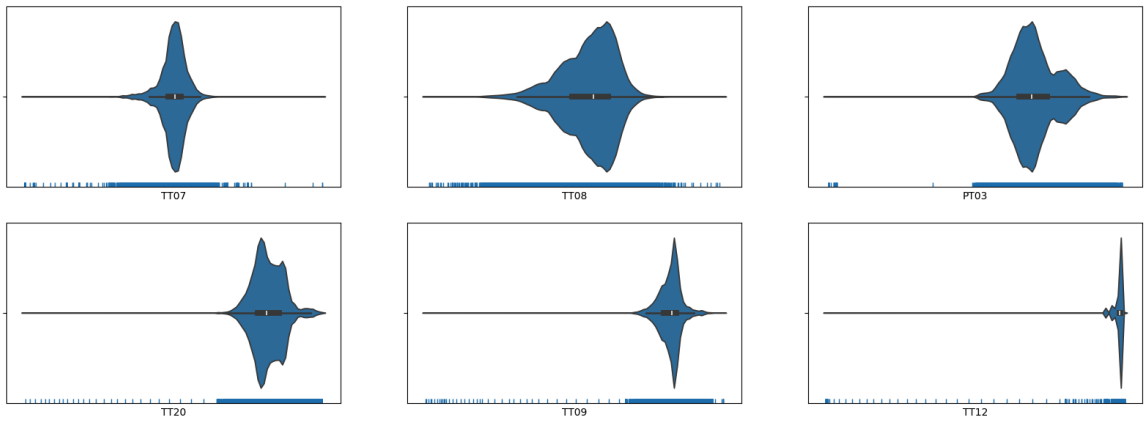
Figur 5.5: Histogram over fordelingen av variabel "Mw" etter tilhørende enzymtype.

Fiolinplott

Fiolinplott var benyttet til å visualisere hvordan data i ulike variabler var fordelt. Visualiseringen, i kombinasjon med teppeplot, muliggjorde inspeksjon av mulige avvik fra gjennomsnittlige verdier. Figur 5.6 viser til fiolinplott av et utvalg av styringsparametere (TT07, TT08, PT03, TT20, TT09, TT12) som datasettet inneholdt.

Blant de seks variablene i figuren, viser det til verdier som avviker betydelige fra gjennomsnittet. Det antyder til mulige ekstremverdier i datasettet. Fordelingen i variablene TT07 og TT08 er nærmest sentralisert og mulige ekstremverdier er tydeligere på begge ender av spektetet av verdier. De resterende parametere har fordelinger der verdiene er konsentrert på høyre side. Fordelingene er dermed venstreskjeve, som typer på mulige avvik og ekstremverdier av flere lavere verdier. Variabel PT03 ser ut til ha identifiserbare verdier som avviker betydelig fra de resterende verdier i variabelen.

Variablene TT20 og TT12 skiller seg av temperaturmålingene, ved å ha en multimodal fordeling med flere topper. Det tyder på at styringsparameterne har flere verdier som innstilles under prosessen. Hovedparten av data i variabel TT12 er forskjøvet til høyere verdier. De lavere verdiene avviker betydelig fra de resterende verdier, og antyder til å kunne være ekstremverdier. En slik ekstrem venstreskjev fordeling for en styringsparameter, kan tyde på en målefeil eller bestemt type avvik i prosessen. Dersom disse verdiene opptrer sammenhengende, kan være tegn til systematisk feil i en begrenset periode.



Figur 5.6: Fiolinplott av kontrollmålinger "TT07", "TT08", PT03", "TT20", "TT09" og "TT12", med markeringer på observasjoner.

5.2 Evaluering av modeller på datasett

I denne seksjonen vil de presenteres resultater fra de ulike maskinlæringsmodellene basert på både veiledet og semi-veiledet læringsmetoder. Det vil vises til resultater av evalueringer på hele datasett med hver av kvalitetsmålingene"Mw", "Smallmolecules" og "Brix-adjusted" som responsvariabel. Hele datasettet inkluderer informasjonen knyttet til alle enzymtypene. Evalueringen av modellene ble utført på et datasett med en ny dimensjon på (28 701, 16). Reduksjonen ble utført etter behandling av opprinnelig datasett, som er forklart videre i Vedlegg A.4.6.

Resultatene i seksjonen vil være oppdelt i flere kategorier. For det første er datasettet oppdelt tre etter typen kvalitetsmåling. De tre datasettene deler samme data for forklaringsvariablene, men har hver sin unike responsvariabel. Seksjon 4.3.3, viser til de to forskjellige strategiene for fordeling av treningssett og testsett som er utviklet. Hver strategi har et sett med egenskaper som vil ha en innvirkning på hvilken type informasjon modellen lærer fra og evalueres på. Resultatene vil dermed vises, med hensyn til gjeldende strategi.

Resultatene presenterer for øvrig sammenlikninger av modellytelsen på datasettene, med og uten variablene med NIR-data. Det vil gi en innsikt i hvordan inkludering av informasjon fra spektroskopisk data påvirker modellens ytelse.

Det er verdt å merke at evalueringsmetrikken RMSE vil variere for hver responsvariabel, ettersom de inneholder verdier av ulike størrelser. Derfor er ikke RMSE-scorene sammenlignbare på tvers av de forskjellige responsvariablene.

5.2.1 Testresultater med Alternativ 1: Enzymtyper

Strategien for fordeling av treningssett og testsett tok hensyn til at fordelingene mellom enzymtypene i hele datasettet var lik i hvert delsett. Tabellene nedenfor presenterer de gjennomsnittlige resultatene på denne type testdata for modellene. Resultatene er av en 10-foldet kryssvalideringsprosess med to repetisjoner, på hele det avsatte treningssettet. De radene som er markert i oransje er veiledede modeller, mens de i grønn er semi-veiledede modeller. Hver modell er den beste kombinasjonen fra optimaliseringsprosessen. For de semi-veiledede modellene er scoren den laveste blant alle undersøkelser som har blitt gjennomført for ulike andeler av umarkerte data.

Tabellen er sortert etter stigende RMSE-score. Antall "*" etter modellnavn beskriver andelen umarkerte data relativ til markert data, som har blitt tilgjengeligjort under treningsfasen. * - 50 %, ** - 100 %, *** - 200 %. Ved **** er all tilgjengelig umarkerte data blitt tilgjengeliggjort.

Tabell 5.1 viser til benchmarks for alle kvalitetsmålingene.

Tabell 5.1: Oversikt over ulike evalueringsresultater og differanse mellom RMSE og MAE for kvalitetsmålinger på testdata for lternativ 1:Enzymtype

Modell	**RMSE(± std)**	**R^2**	**MAPE (%)**	$\frac{(\mathbf{RMSE-MAE})}{\mathbf{MAE}}(\%)$
Mw	785.34(± 2.0937)	0.89	8.63	38.40
BrixAdjusted	0.01(± 0.0000)	0.53	8.19	52.44
SmallMolecules	0.84(± 0.0057)	0.68	5.69	34.26

Hele datasett, Alternativ 1 - Mw

Tabell 5.2: Oversikt over ulike evalueringsresultater og differanse mellom RMSE og MAE for hele datasettet med kvalitetsmåling "Mw" som respons. Alternativ 1: Enzymtype, er benyttet som strategi for fordeling av trening og testsett.

Modell	**RMSE (± std)**	**R^2**	**MAPE (%)**	$\frac{(\mathbf{RMSE-MAE})}{\mathbf{MAE}}(\%)$
RFR	548,64 (± 13,00)	0,95	5,80	40,12
Selv_RFR****	554,10 (± 25,55)	0,95	5,80	40,27
SVR	582,25 (± 4,59)	0,94	6,30	38,62
KNR	618,67 (± 11,90)	0,93	6,10	45,97
Coreg**	629,47 (± 19,26)	0,93	6,20	48,11
BHD - Domene**	974,10 (± 12,31)	0,83	10,50	31,82

Tabell 5.2 presenterer resultater for datasett med Mw som responsvariabel. Her viser den at begge trebaserte modellene RFR og Selv_RFR generelt presterer bedre enn øvrige modeller. Utenom på det relative avviket mellom RMSE og MAE, indikerer de resterende metrikkene på at modellene oppnår høyere prediksjonsnøyaktighet enn resterende modeller. Blant de to modellene har den veiledede varianten av RFR en marginal fordel, da standardavviket i RMSE-scoren er betydelig lavere enn den semi-veiledede varianten. Det er verdt å merke at Selv_RFR har det høyeste standardavviket blant de evaluerte modellene.

Når det gjelder standardavviket på RMSE-scorene, viser de veiledede modellene en tendens til lavere variasjon sammenlignet med de semi-veiledede modellene. Spesielt har SVR det laveste standardavvikent blant modellene.

Modellen BHD-Domene skiller seg spesielt ut med dårligst modellprestasjon. Den oppnår den lavste gjennomsnittlige RMSE-scoren og R^2-verdien med god margin, samt høyste MAPE-verdien. Imidlertid viser BHD-Domene den beste prestasjonen når det gjelder det relative avviket mellom RMSE og MAE. Det er en indikasjon på at modellen er mindre følsom fior utliggere og ekstreme verdier sammenlignet med de andre evaluerte modellene.

Hele datasett, Alternativ 1 - SmallMolecules

Tabell 5.3: Oversikt over ulike evalueringsresultater og differanse mellom RMSE og MAE for hele datasettet med kvalitetsmåling "SmallMolecules" som respons. Alternativ 1: Enzymtype er benyttet som strategi for fordeling av trening og testsett.

Modell	**RMSE (± std)**	**R^2**	**MAPE (%)**	$\frac{(\mathbf{RMSE-MAE})}{\mathbf{MAE}}(\%)$
Selv_RFR ***	0,608 (± 0,01)	0,83	4,10	24,73
RFR	0,631 (± 0,01)	0,82	4,20	36,58
KNR	0,651 (± 0,02)	0,81	4,20	33,04
Coreg***	0,688 (± 0,02)	0,79	4,60	24,94
SVR	0,710 (± 0,02)	0,77	4,50	36,13
BHD-Domene ***	1,032 (± 0,01)	0,52	7,50	25,62

Tabell 5.3 viser resultater av modellene som presterte for hele datasettet med kvalitetsmåling "SmallMolecules" som responsvariabel. Her ble det benyttet alternativ 1 for fordeling av trening- og testdata. Tabellen viser at begge RFR modellene fra hver sin respektive læring predikerte med lavest RMSE-score, i tillegg til at standardavviket er relativt lavt sammenliknet med de andre modellene. Det er små forskjeller mellom alle modellene som har prestert på denne responsvariabelen. Videre blir det vist en gradvis stigende økning i alle MAPE-verdiene i tabellen for alle modellene.

BHD-modellen presterte dårligst med høyest RMSE-score. Standardavviket er relativt lavt, og det viser til at den overordnet har jevn predikeringsevne. Denne modellen er i tillegg nesten like robust som den beste modellen, selv_rfr, mot utliggere og støy i datasettet.

Hele datasett, Alternativ 1 - BrixAdjusted

Tabell 5.4: Oversikt over ulike evalueringsresultater og differanse mellom RMSE og MAE for hele datasettet med kvalitetsmåling "BrixAdjusted" som respons. Alternativ 1: Enzymtype, er benyttet som strategi for fordeling av trening og testsett.

Modell	**RMSE (± std)**	**R^2**	**MAPE (%)**	$\frac{(\mathbf{RMSE-MAE})}{\mathbf{MAE}}$(%)
Coreg *	0,009 (±0,000)	0,75	6,00	50,0
KNR	0,009 (±0,000)	0,74	6,10	50,0
RFR	0,010 (±0,000)	0,73	6,00	66,7
Selv_RFR *	0,010 (±0,000)	0,72	6,10	66,7
BHD - Domene **	0,013 (±0,000)	0,51	8,70	44,4
SVR	0,015 (±0,000)	0,37	11,10	25,0

Tabell 5.4 presenterer prediksjonen til modellene for kvalitetsmåling "BrixAdjusted" som responsvariabel. Coreg er modellen som presterer best med lavest RMSE-score. Coreg og KNR har lik RMSE, med minimale differanser i R^2 verdi.

Tabell 5.4 viser en oversikt over modellene som predikerte på hele datasettet med kvalitetsmåling "BrixAdjusted" som responsvariabel. Alternativ 1 for fordeling av trening- og testdata ble det benyttet. Det blir presentert at semi-veiledet Coreg presterer med best evne. Selv om RMSE-score og avviket er lavt, ser vi den lille forskjellen i R^2 verdien. Deretter kommer KNR modellen som er nokså lik Coreg med minimale forskjeller, som synes ved å sammenlikne R^2-scorene og MAPE verdiene. Det er minimale forskjeller mellom nesten alle modeller i denne tabellen.

SVR-modellen presterte dårligst, med høyest RMSE-score og betydelig mye lavere R^2-score sammenliknet med de andre modellene. Til tross for dette indikerer "RMSE / MAE (%)" verdien til denne modellen at den er mer robust mot støy og utliggere sammenliknet med de andre modellene som ble trent på dette treningssettet.

Datasett med og uten NIR

Følgende kapittel vil det presenteres tabeller der modellprestasjonene blir sammenliknet med og uten data fra en NIR-sensor. Det presenteres tre tabeller. Hver tabell vil fokusere på en kvalitetsmåling hver. Rekkefølgen på tabellene vil være; Mw, SmallMolecules og BrixAdjusted.

Alternativ 1 (NIR) - Mw

Tabell 5.5: Oversikt over ulike evalueringsresultater og differanse mellom RMSE og MAE i (%) for ulike modeller med og uten NIR på Mw med Alternativ 1.

Modell	**RMSE (± std)**	**R^2**	**MAPE (%)**	$\frac{(\mathbf{RMSE-MAE})}{\mathbf{MAE}}$(%)
RFR	548,64 (± 13,00)	0,95	5,80	40,12
SVR	582,25 (± 4,59)	0,94	6,30	38,62
KNR	618,67 (± 11,90)	0,93	6,10	45,97
RFR (∼ NIR)	771,52 (± 14,82)	0,90	8,80	30,93
KNR (∼ NIR)	856,86 (± 30,67)	0,87	9,10	36,04
SVR (∼ NIR)	1 007,66 (± 7,86)	0,82	11,20	33,69

Tabell 5.5 presenterer prediksjonen til de veiledede modellene som har predikert på datasettet, med og uten NIR verdier kvalitetsmåling "Mw" som responsvariabel. Det blir vist at RFR med NIR variabler presterer best og har lavest RMSE-score og høyest R^2 verdi. SVR har lavest standardavvik i RMSE-verdiene. Tabellen viser et klart skille der modellene presterer bedre med NIR målingene.

Alternativ 1 (NIR) - SmallMolecules

Tabell 5.6: Oversikt over ulike evalueringsresultater og differanse mellom RMSE og MAE i (%) for ulike modeller med og uten NIR på Smallmolecules med Alternativ 1.

Modell	**RMSE (± std)**	**R^2**	**MAPE (%)**	$\frac{(\mathbf{RMSE-MAE})}{\mathbf{MAE}}$(%)
RFR	0,63 (± 0,01)	0,82	4,20	39,51
KNR	0,65 (± 0,02)	0,81	4,20	36,58
SVR	0,71 (± 0,02)	0,77	4,50	44,02
RFR (∼ NIR)	0,90 (± 0,01)	0,64	6,20	27,68
KNR (∼ NIR)	0,90 (± 0,02)	0,63	6,40	31,23
SVR (∼ NIR)	0,98 (± 0,01)	0,57	6,10	43,32

Tabell 5.6 presenterer prediksjonen til de veiledede modellene som har predikert på datasettet, med og uten NIR verdier kvalitetsmåling "SmallMolecules" som responsvariabel.

Det blir vist at RFR med NIR variabler presterer best og har lavest RMSE-score og høyest R^2 verdi. SVR uten NIR målinger høyest RMSE-score og lavest R^2 verdi.

Alternativ 1 (NIR) - BrixAdjusted

Tabell 5.7: Oversikt over ulike evalueringsresultater og differanse mellom RMSE og MAE i (%) for ulike modeller med og uten NIR for BrixAdjusted med Alternativ 1.

Modell	**RMSE (± std)**	**R^2**	**MAPE (%)**	$\frac{(\mathbf{RMSE}-\mathbf{MAE})}{\mathbf{MAE}}(\%)$
KNR	0,009 (± 0,000)	0,74	6,10	50,00
RFR	0,010 (± 0,000)	0,73	6,00	66,67
KNR (∼ NIR)	0,010 (± 0,000)	0,70	6,80	42,86
RFR (∼ NIR)	0,011 (± 0,000)	0,67	6,80	83,33
SVR	0,015 (± 0,000)	0,37	11,10	25,00
SVR (∼ NIR)	0,025 (± 0,001)	-0,86	17,10	25,00

Tabell 5.7 presenterer prediksjonen til de veiledede modellene som har predikert på datasettet, med og uten NIR verdier kvalitetsmåling "BrixAdjusted" som responsvariabel. Det blir vist at KNR med NIR variabler presterer best og har lavest RMSE-score og høyest R^2 verdi, men viser til at modellen er følsom mot utliggere. SVR uten NIR målinger viser til en negativ R^2 verdi.

Flere evalueringsresultater

Tabell 5.8 viser resultater fra evalueringer fra kryssvalideringen for datasettet med Mw som respons. Det er lagt ved paramtere av interesse for hver modell. Differansen mellom RMSE-scorene i trening og test data er betydelig hos COREG og BHD, noe som kan tyde på overtilpasning. Det relative standardavviket i "Selv_RFR" er nær grenseverdien som var satt til optimaliseringen, på 5 %

Tabell 5.8: Oversikt over RMSE med standardavvik for både test og treningsett for ulike modeller med og uten NIR for Mw Alt 1.

Modell	**Interessante parametere**	**RMSE test (± std)**	**RMSE trening (± std)**
RFR	n_estimators: 245	548,64 (± 13,00)	202,387 (± 4,808)
Selv_RFR ****	n_estimators: 433, std_terskel: 0,045	554,10 (± 25,55)	259,12 (± 12,174)
SVR	epsilon: 25,249	582,25 (± 4,59)	621,073 (± 16,383)
KNR	n_neighbors: 5	618,67 (± 11,90)	0,00 (± 0)
Coreg **	k1: 7, k2: 7	629,47 (± 19,26)	219,692 (± 8,469)
RFR (∼ NIR)	n_estimators: 454	771,52 (± 14,82)	290,734 (± 7,345)
KNR (∼ NIR)	n_neighbors: 4	856,86 (± 30,67)	0,00 (± 0)
BHD - Domene *	alpha: 0,174	974,10 (± 12,31)	76,985 (± 2,427)
SVR (∼ NIR)	epsilon: 92,542	1007,66 (± 7,86)	1051,673 (± 12,966)

5.2.2 Testresultater med Alternativ 2: Dag og kontinuitet

Tabell 5.9 viser til benchmarks for alle kvalitetsmålingene.

Tabell 5.9: Oversikt over ulike evalueringsresultater og differanse mellom RMSE og MAE for kvalitetsmålinger på testdata for Alternativ 2

Modell	**RMSE(± std)**	**R^2**	**MAPE (%)**	$\frac{(\mathbf{RMSE-MAE})}{\mathbf{MAE}}(\%)$
MW	912.72(± 49.2854)	0.82	8.16	42.56
SmallMolecules	0.81(± 0.0284)	0.66	5.39	31.22
BrixAdjusted	0.01(± 0.0001)	0.65	7.40	33.75

Alternativ 2 - Mw

Tabell 5.10: Oversikt over ulike evalueringsresultater og differanse mellom RMSE og MAE i (%) for hele datasettet med kvalitetsmåling "Mw" som respons. Alternativ 2: Dag og kontinuitet, er benyttet som strategi for fordeling av trening og testsett.

Modell	**RMSE (± std)**	**R^2**	**MAPE (%)**	$\frac{(\mathbf{RMSE-MAE})}{\mathbf{MAE}}(\%)$
SVR	614,96 (± 22,57)	0,92	5,90	36,60
RFR	659,74 (± 20,82)	0,90	6,30	36,30
Selv_RFR*	663,26 (± 18,58)	0,90	6,30	34,00
Coreg****	676,83 (± 23,23)	0,90	6,40	44,20
KNR	699,01 (± 14,46)	0,90	6,70	42,30
BHD - Domene****	747,41 (± 33,25)	0,89	6,90	42,40

Tabell 5.10 presenterer prediksjonen til modellene for kvalitetsmåling "Mw" som responsvariabel. SVR er modellen best nøyaktighet i form av lavest RMSE-score. BHD har dårligst nøyaktighet med høyest RMSE verdi.

Alternativ 2 - SmallMolecules

Tabell 5.11: Oversikt over ulike evalueringsresultater og differanse mellom RMSE og MAE i (%) for hele datasettet med kvalitetsmåling "SmallMolecules" som respons. Alternativ 2: Dag og kontinuitet, er benyttet som strategi for fordeling av trening og testsett.

Modell	**RMSE (± std)**	**R^2**	**MAPE (%)**	$\frac{(\mathbf{RMSE-MAE})}{\mathbf{MAE}}(\%)$
SVR	0,674 (± 0,01)	0,76	4,50	27,89
Selv_RFR ****	0,731 (± 0,02)	0,72	4,90	29,38
RFR	0,758 (± 0,02)	0,70	5,10	28,91
BHD - Domene ****	0,797 (± 0,03)	0,67	5,30	33,50
Coreg ****	0,798 (± 0,03)	0,67	5,60	26,27
KNR	0,851 (± 0,04)	0,62	5,90	25,70

Tabell 5.11 presenterer prediksjonen til modellene for kvalitetsmåling "SmallMolecules" som responsvariabel. SVR har lavest RMSE verdi, og god tilpasningsevne med en høy R^2 score. KNR har dårligst nøyaktighet med høyest RMSE verdi.

Alternativ 2 - BrixAdjusted

Tabell 5.12: Oversikt over ulike evalueringsresultater og differanse mellom RMSE og MAE i (%) for hele datasettet med kvalitetsmåling "BrixAdjusted" som respons. Alternativ 2: Dag og kontinuitet, er benyttet som strategi for fordeling av trening og testsett.

Modell	**RMSE (± std)**	**R^2**	**MAPE (%)**	$\frac{(\mathbf{RMSE-MAE})}{\mathbf{MAE}}(\%)$
RFR	0,006 (± 0,000)	0,88	4,90	20,00
KNR	0,006 (± 0,000)	0,88	5,00	20,00
Selv_RFR*	0,006 (± 0,000)	0,88	13,80	20,00
Coreg****	0,006 (± 0,000)	0,87	4,90	20,00
BHD - Domene*	0,010 (± 0,000)	0,70	7,10	20,00
SVR	0,022 (± 0,001)	-0,46	5,20	37,00

Tabell 5.12 presenterer prediksjonen til modellene for kvalitetsmåling "BrixAdjusted" som responsvariabel. RFR og KNR har like lav RMSE verdi, men skiller seg når MAPE beregnes. SVR predikerer verst som blir vist til med en høy RMSE verdi. I tillegg har

den en negativ R^2 verdi som viser til dårlig modell tilpasning. Denne modellen er også minst robust mot utliggere.

Datasett med og uten NIR

Alternativ 2 (NIR) - Mw

Tabell 5.13: Oversikt over ulike evalueringsresultater og differanse mellom RMSE og MAE i (%) for ulike modeller med og uten NIR for Mw med Alternativ 2.

Modell	**RMSE (± std)**	**R^2**	**MAPE (%)**	$\frac{(\mathbf{RMSE-MAE})}{\mathbf{MAE}}$(%)
RFR (∼ NIR)	604,15 (± 11,01)	0,92	6,60	24,50
SVR	614,06 (± 22,57)	0,92	5,90	36,60
RFR	659,74 (± 20,82)	0,90	6,30	36,30
KNR	699,01 (± 14,46)	0,90	6,70	42,30
KNR (∼ NIR)	729,68 (± 16,21)	0,88	7,60	28,60
SVR (∼ NIR)	739,53 (± 55,92)	0,88	7,20	35,60

Tabell 5.13 presenterer prediksjonen til de veiledede modellene som har predikert på datasettet, med og uten NIR verdier kvalitetsmåling "Mw" som responsvariabel. Det blir vist at RFR uten NIR variabler presterer best og har lavest RMSE-score og høyest R^2 verdi. SVR uten NIR målinger høyest RMSE-score og lavest R^2 verdi.

Alternativ 2 (NIR) - SmallMolecules

Tabell 5.14: Oversikt over ulike evalueringsresultater og differanse mellom RMSE og MAE i (%) for ulike modeller med og uten NIR for SmallMolecules med Alternativ 2.

Modell	**RMSE (± std)**	**R^2**	**MAPE (%)**	$\frac{(\mathbf{RMSE-MAE})}{\mathbf{MAE}}(\%)$
SVR	0,674 (± 0,01)	0,76	4,50	27,89
RFR	0,758 (± 0,02)	0,70	5,10	28,91
SVR (∼ NIR)	0,814 (± 0,03)	0,65	5,50	30,24
KNR	0,851 (± 0,04)	0,62	5,90	25,70
RFR (∼ NIR)	0,896 (± 0,02)	0,58	6,20	23,59
KNR (∼ NIR)	0,947 (± 0,03)	0,53	6,30	29,37

Tabell 5.14 presenterer prediksjonen til de veiledede modellene som har predikert på datasettet, med og uten NIR verdier kvalitetsmåling "SmallMolecules" som responsvariabel. Det blir vist at SVR med NIR variabler presterer best og har lavest RMSE-score og høyest R^2 verdi. KNR uten NIR har høyest RMSE-score og lavest R^2 verdi.

Alternativ 2 (NIR) - BrixAdjusted

Tabell 5.15: Oversikt over ulike evalueringsresultater og differanse mellom RMSE og MAE i (%) for ulike modeller med og uten NIR for BrixAdjusted med Alternativ 2.

Modell	**RMSE (± std)**	**R^2**	**MAPE (%)**	$\frac{(\mathbf{RMSE-MAE})}{\mathbf{MAE}}(\%)$
RFR	0,006 (± 0,000)	0,88	4,90	20,00
KNR	0,006 (± 0,000)	0,88	5,00	20,00
RFR (∼ NIR)	0,007 (± 0,000)	0,87	5,40	37,50
KNR (∼ NIR)	0,008 (± 0,001)	0,83	6,00	33,33
SVR	0,022 (± 0,001)	-0,46	5,20	40,00
SVR (∼ NIR)	0,022 (± 0,001)	-0,46	13,80	37,50

Tabell 5.15 presenterer prediksjonen til de veiledede modellene som har predikert på datasettet, med og uten NIR verdier kvalitetsmåling "BrixAdjusted" som responsvariabel. Det blir vist at RFR og KNR med NIR variabler presterer best og har laveste RMSE-scorene og høyest R^2 verdi. Det er minimale forskjeller i MAPE score mellom begge modellene. SVR uten NIR målinger høyest RMSE-score og lavest R^2 verdi.

Flere evalueringsresultater

Tabell 5.16 viser resultater fra evalueringer fra kryssvalideringen for datasettet med Mw som respons for Alternativ 2:Dag og kontinuitet som oppdelingsstrategi. Det relative standardavviket i "Selv_RFR" er nær grenseverdien er lavere på 3,5 %

Tabell 5.16: Oversikt over RMSE med standardavvik for både test og treningsett for ulike modeller med og uten NIR for Mw Alternativ 2.

Modell	**Interessant parameter**	**RMSE test (± std)**	**RMSE trening (± std)**
RFR (∼ NIR)	n_estimators: 468	604,15 (± 11,01)	305,547 (± 8,182)
SVR	epsilon: 0,088	614,06 (± 22,57)	618,506 (± 16,138)
RFR	n_estimators: 238	659,74 (± 20,82)	212,836 (± 5,081)
Selv_RFR *	n_estimators: 417, std_terskel: 0,034	663,26 (± 18,58)	204,502 (± 3,412)
Coreg ****	k1: 7, k2: 2	670,83 (± 23,23)	211,926 (± 6,788)
KNR	n_neighbors: 2	699,01 (± 14,46)	0 (± 0)
KNR (∼ NIR)	n_neighbours: 6	729,68 (± 16,21)	0 (± 0)
SVR (∼ NIR)	epsilon: 60,362	739,53 (± 55,92)	1082,847 (± 22,805)
BHD - Domene ****	alpha: 0,021	747,41 (± 33,25)	50,919 (± 0,514)

5.2.3 Undersøkelser på datasett med Mw som respons

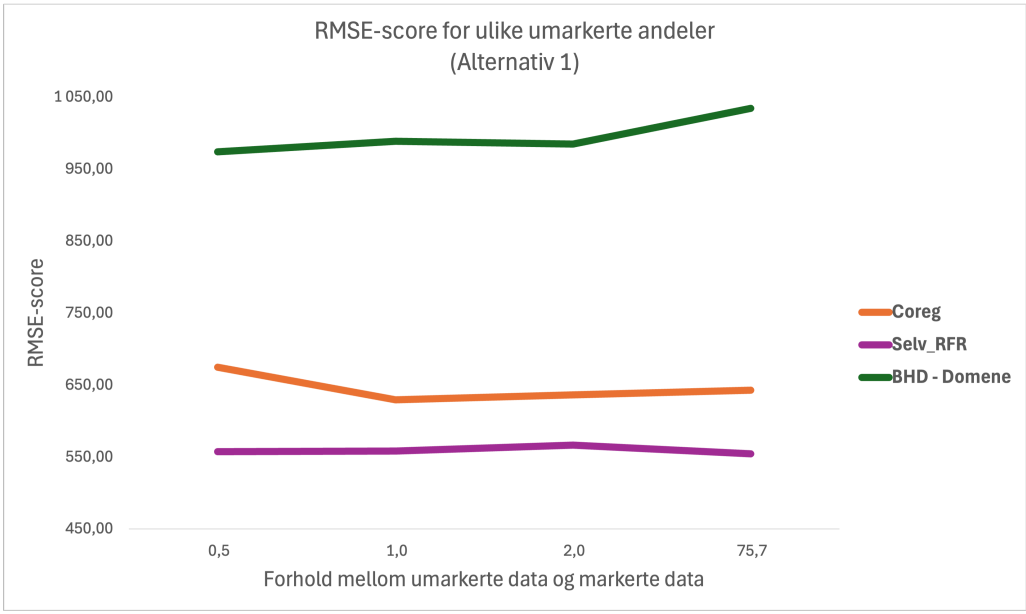
Følgende seksjon vil presentere resultatet av flere undersøkelser gjort på modellering på datasettet med Mw som respons. Ulike modeller har blitt optimalisert og modellert på ulike andeler av data, for evalueres på samme avsatte testdata. Det vil bli vises til grafer for evaluering på testdata basert på modelloppbygning av ulike mengder av både markerte og umarkerte data. Ved grafer over utvikling prediksjonsytelse ved inkludering av ulike mengder med umarkerte data, vil det vises til de optimaliserte semi-veiledede modeller for hver andel.Evalueringen av modellene er basert på en gjennomsnitt RMSE-score fra kryssvalidering, der de optimaliserte modellene var trent på hele det opprinnelige treningssettet.

Ved grafer som illustrerer utviklingen av prediksjonsytelse med ulike mengder markerte treningsdata, vises det til optimaliserte modeller av et begrenset utvalg algoritmer. Disse er valgt med hensyn til å vise de meste interessante funn. Det må merkes at andeler av markerte treningsdata beskriver mengden av opprinnelig treningsdata som var gjort tilgjengelig under optimalering av modellene. Informasjonen i grafen representerer gjennomsnittlig RMSE-score. I tillegg representeres standardavviket i RMSE-score som størrelsen på sirkelen. Det må merkes at størrelsen er relativ til standardavviket til de andre modellene i grafen.

For evaluering av modeller optimalert for ulike andeler av markerte data gjort tilgjengelig,blir to veiledede og to semi-veiledede algoritmer presentert. I grafene som er utvalgt vises det til semi-veiledede modeller som er blitt trent på del av umarkerte data som tilsvarer 200 % av mengde markerte data under optimalisering. Evaluering av RMSE-score er dermed basert på samme opprinnelige markerte treningsdata, men antall umarkerte data er avhengig av markerte data under optimaliseringen. Veiledede modellene er optimalisert kun på ulike andelene av markerte data og evaluert basert på opprinnelig treningsdata.

Ulike umarkerte andeler med Alternativ 1: Enzymtype

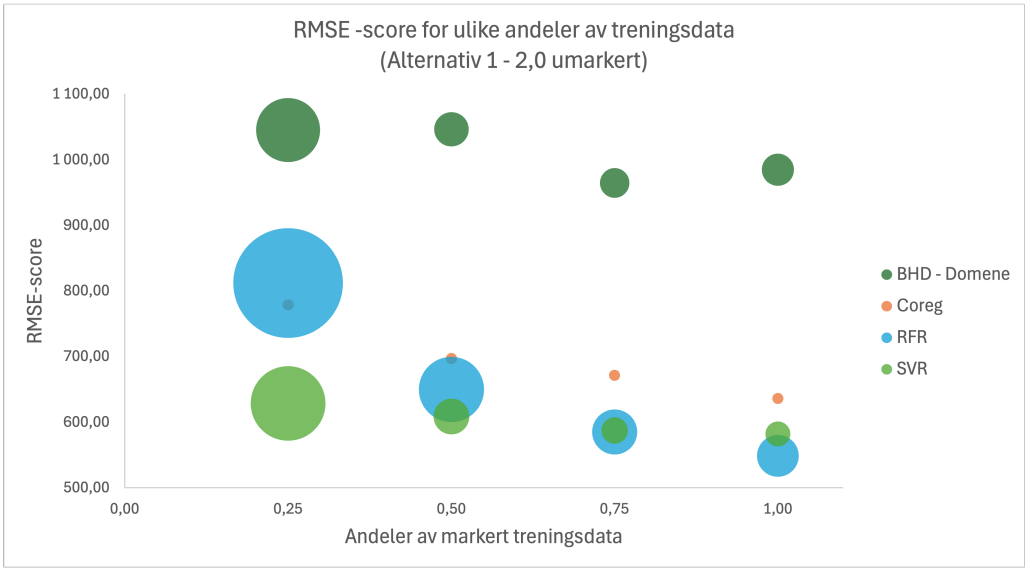
Figur 5.7 er en graf som viser utviklingen av gjennomsnittlig RMSE-score av semi-veiledede modeller trent på ulike umarkerte andeler av data. Utvalget av umarkerte data var basert på at fordelingen av enzymtyper var lik i de umarkerte data som i det markerte data. Grafen viser at modellen "Selv_RFR" presterte best av alle modellene, med en stabil RMSE-score over de ulike andelene. Modellen som predikerte dårligst var BHD, med gjennomsnittlig RMSE-score over 950 over alle andeler. Grafen viser en stabil utvikling for alle algoritmer, før en oppgaven i RMSE-score når all tilgjengelig umarkerte data blir gitt til modellene til opplæring.



Figur 5.7: Denne figuren viser gjennomsnittlig RMSE-score for 3 semi-veiledede regresjonsmodeller for ulike andeler av umarkerte data for Mw som respons. Alternativ 1:Enzymtype er valgt oppdelignsstrategi.

Ulike markerte andeler og umarkerte andel med Alternativ 1: Enzymtype

Figur 5.8 visualiserer utviklingen av 4 algoritmer som har blitt evaluert basert på optimalisering av 4 forskjellige andeler av markert data. Det vises til en trend der standardavviket til RFR modeller reduserer betydelig fra 25 % til 100 % markerte data. Ved 100 % presterer en RFR modell bedre enn andre modeller. Modeller av COREG algoritmen ser ut til å robuste prediksjoner med lavere standardavvik i RMSE-score over alle markerte treningsandeler. Modeller av BHD presterer dårligst med høyest RMSE-score ved alle tilfellene, med varierende størrelse på standardavviket.

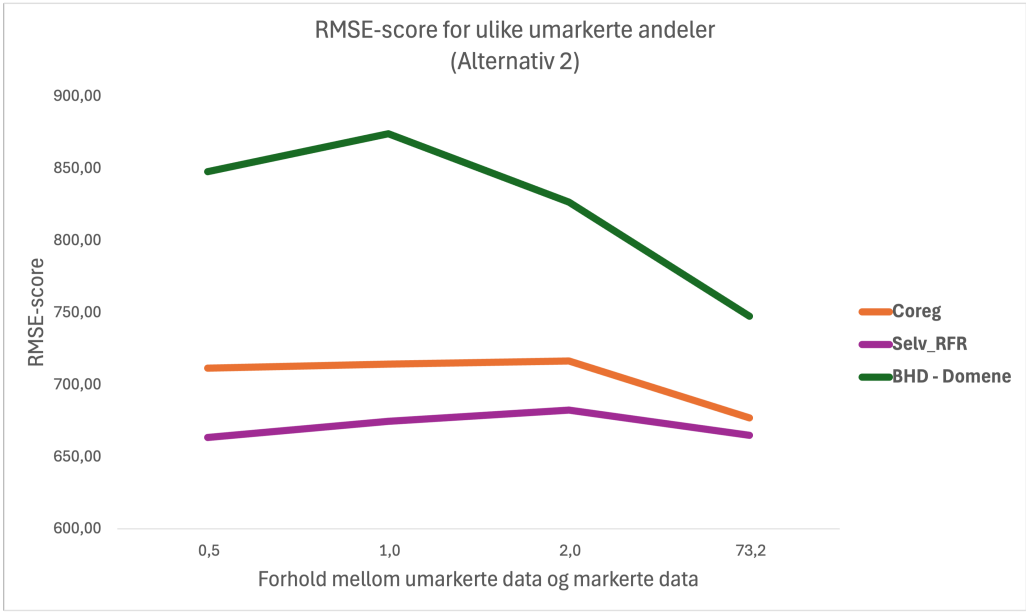


Figur 5.8: Denne figuren viser gjennomsnittlig RMSE-score for ulike modeller av 4 algoritmer over forskjellige andeler av markerte data under optimalisering. Størrelsen på sirklene i grafen representerer standardavviket i RMSE-score. Oppdelingsstrategi var Alternativ 1:Enzymtype.

Ulike umarkerte andeler med Alternativ 2: Dag og kontinuitet

Figur 5.9 viser til utviklingen av gjennomsnittlig RMSE-score av semi-veiledede modeller trent på ulike umarkerte andeler av data. I dette tilfellet er det ikke tatt hensyn til at fordelingen av enzymtyper i de umarkerte data skal være lik som i markerte data. for ulike mengder umarkerte data. Grafen viser at modellenene basert på "Selv_RFR" presterte bedre blant modellene. BHD modellene presterte gjennomgående dårligere. Utviklingen av RMSE-score ser ut til å øke blant alle modeller fram til $x = 2.0$. Deretter synker RMSE-scoren for modellene for alle algortimene.

.

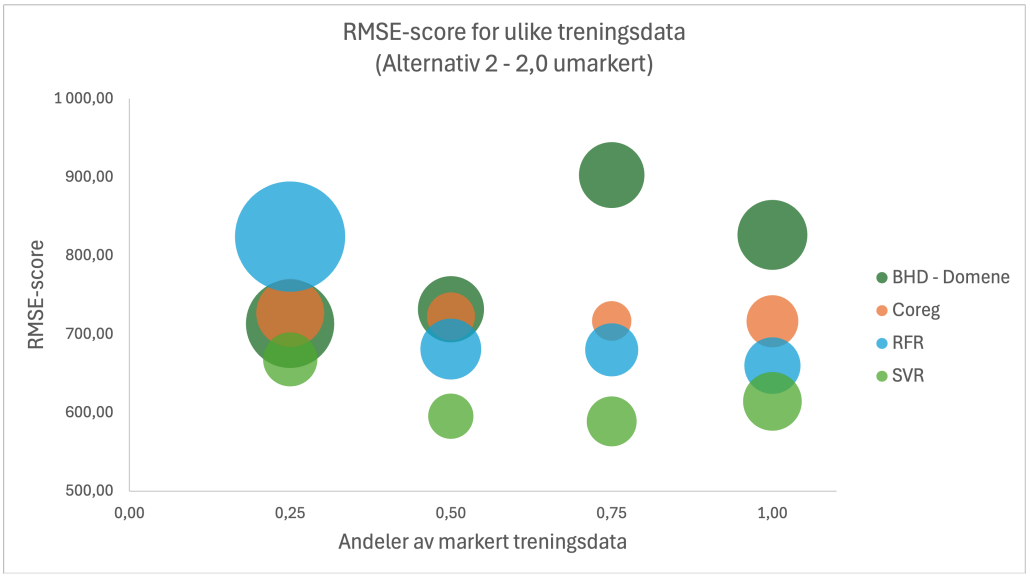


Figur 5.9: Denne figuren viser gjennomsnittlig RMSE-score for 3 semi-veiledede regresjonsmodeller for ulike andeler av umarkerte data for Mw som respons. AAlternativ 2:Dag og kontinuitet er valgt oppdelignsstrategi.

Ulike markerte andeler og umarkerte andel med Alternativ 1: Dag og kontinuitet

Figur 5.8 visualiserer utviklingen av 4 algoritmer som har blitt evaluert basert på optimalisering av 4 forskjellige andeler av markert data. Det vises til en trend der standardavviket til RFR modeller reduserer betydelig fra 25 % til 100 % markerte data. Ved 100 % presterer en RFR modell bedre enn andre modeller. Modeller av COREG algoritmen ser ut til å robuste prediksjoner med lavere standardavvik i RMSE-score over alle markerte treningsandeler. Modeller av BHD presterer dårligst med høyest RMSE-score ved alle tilfellene, med varierende størrelse på standardavviket.

Figur 5.10 visualiserer utviklingen av 4 algoritmer som har blitt evaluert basert på optimalisering av 4 forskjellige andeler av markert data. Det er minimale forskjeller i mellom standaridavvikene i RMSE-score blant de ulike modellene. SVR-modeller presterte bedre med lavere RMSE-score over alle markerte treningsandeler, spesielt ved $x = 0.5$ og $x = 0.75$. Prestasjonen av COREG-modellene var stabil over de ulike andelene og forble nærmest uendret. Forskjellen i RMSE-score er lavere ved mindre markerte treningsandeler på $x = 0.25$. Modeller av BHD så ut til å prestere på ulik linje av andre modeller ved lavere markerte treningsandler. Ved $x = 0.25$ var RMSE-scoren og standardavviket bedre i forhold til en RFR-modell. For $x = 0.75$ og $x = 1.0$ var BHD-modellene de med svakest prestasjonen.



Figur 5.10: Denne figuren viser gjennomsnittlig RMSE-score for ulike modeller av 4 algoritmer over forskjellige andeler av markerte data under optimalisering. Størrelsen på sirklene i grafen representerer standardavviket i RMSE-score. Oppdelingsstrategi var Alternativ 2: Dag og kontinuitet.

5.3 MCDA-analyse: Resultater

For å etablere en grundigere og bredere sammenligning mellom mulig implementering av datadreven soft-sensor modell basert på forskjellige maskinlæringer og inkludering av spektroskopisk sensor, er det valgt å utføre en MCDA-analyse. Resultatene tar hovedsakelig utgangspunkt i evalueringene gjort på trening og testdata på de forskjellige algoritmene. I tillegg er det valgt å inkludere økonomiske aspekter ved mulig implementering, som krav til mengde data og investering i sensor. Vurderinger for valg av alternativer basert på metoden presentert i Seksjon 4.4.2. Beregning av totalscore (Formel 4.2), er basert på kriteriene og vektingene i Tabell 4.9.

Mulighetsstudie var basert på følgende alternativer og er kort oppsummert som følgende:

Dagens prosess

- Produktkvalitet vurderes ikke før i ettertid i produksjonen, og det gjøres ingen beslutninger basert på grad av kvalitet på produktet underveis.
- Brukes ikke soft-sensor og maskinlæringsalgorimter til støtte og vurderinger.
- Ingen permanent utplassering av spektroskopisk sensor.

Alternativ 1 - Prosessstyring med veiledede maskinlæringsalgoritmer og NIR-måler

- Implementering av veiledede maskinlæringsalgoritmer for å utvikling en soft-sensor.
- Produktkvalitet kan vurderes underveis i prosessen, og muligheter til beslutninger basert på grad av kvalitet på produktet.
- Krever fullstendig datasett og innsamling av kvalitetsmålinger.
- Forutsetter permanent utplassering av spektroskopisk sensor.

Alternativ 2 - Prosessstyring med semi-veiledede maskinlæringsalgoritmer og NIR-måler

- Implementering av semi-veiledede maskinlæringsalgoritmer for å utvikling en soft-sensor.
- Produktkvalitet kan vurderes underveis i prosessen, og muligheter til beslutninger basert på grad av kvalitet på produktet.
- Krever ikke fullstendig datasett og innsamling av kvalitetsmålinger.
- Forutsetter permanent utplassering av spektroskopisk sensor.

Alternativ 3 - Prosessstyring med maskinlæringsalgoritmer uten NIR-måler

- Implementering av maskinlæringsalgoritmer for å utvikling en soft-sensor.
- Produktkvalitet kan vurderes underveis i prosessen, og muligheter til beslutninger basert på grad av kvalitet på produktet.
- Krever fullstendig datasett og innsamling av kvalitetsmålinger ved veiledede algoritmer. Med semi-veiledede er det ikke krav.
- Ingen permanent utplassering av spektroskopisk sensor.

5.3.1 Begrunnelser for kvalitative score

Prediksjonsytelse og modellkvalitet:

Basert på en helhetlig tilnærming og vurdering av modellenes generelle prestasjoner for evalueringsresultater som RMSE, R^2, MAPE og RMSE/MAE på tvers av kvalitetsmålinger som responsvariabler, gjør veiledede modeller med spektroskopisk data det generelt bedre. Modeller basert på RFR og SVR presterer bedre for flere av kvalitetsmålingene, og indikerer høy nøyaktighet og pålitelighet i prediksjonen til sammenligning med andre modeller. Selv om noen modeller presterer dårligere for BrixAdjusted (5.4) og SmallMolecules (5.3), så viser generelle resultater at veiledede modeller som RFR og SVR oppnår høye score for de fleste responsvariabler. Det gir en sterk indikasjon på at veiledede gir konsistente prediksjoner med god nøyaktighet på tvers av datasett og ulike oppdelingsstrategier for trening og test data. På grunn av god robusthet til tross for ulike kvalitetsmålinger, velges det å gi alternativ 1 en høy score på 4.

Semiveilede modeller viser imidlertid noe lavere ytelse i sammenligning. COREG og Selv_RFR har i tilfeller, som referert i tabellene overfor, vist å prestere god for BrixAdjsuted og SmallMolecules. BHD presterer gjennomgående dårlig derimot gjennomgående dårlig på tvers av datasett. Det viser også til overtilpasning i Tabell 5.8 og Tabell 5.16. Dermed gis alternativ 2 en score på 3.

Dagens situasjon involverer ingen maskinlæringsalgoritmer og baseres på standardiserte styringsparametere uten tilpasning til råmaterialets variasjoner. Siden prosessen ikke kan ta hensyn til variasjoner og spesifikke forhold som kan verken påvirke produktkvalitet eller basere beslutninger i tidligere fase basert på produktkvalitet. Ved fravær av modell, gis alternativet en score på 1.

For alternativ 3 er også gitt en score på 1. Ved sammenligning er det gjennomgående at maskinlæringsalgoritmer uten spektroskopisk data presterer dårligere enn statistiske estimater for kvalitetsmålingene. Prediksjonsytelsen er dermed lavere enn estimater uten modellering.

Funksjonalitet og kompleksitet:

Dagens løsning krever ingen for endringer og implementering av algoritmer som kan være komplekse. Dette alternativet vil både spare tid og ressurser og siden det ikke vil være noe behov vor å endre på noen prosesser i bedriften, vil dette alternativet spare tid og ressurser.
Dagens løsning vil dermed score 4.

Alternativ 1 tar i bruk veiledede modeller. Disse er mindre komplekse i forhold og krever ikke like mye minne med begrenset data. Dersom man ser bort i fra RFR, er de resterende modellene mindre komplekse til sammenlikning. Siden disse modellene har blitt ganske utbredt er gode implementeringer for dette i dag som fører til god brukervennlighet.

Semi-veiledede modeller har mulighet til å håndtere store mengder data, og bruker lengre tid under modelleringsfasen. Under optimeringsfasen brukte modeller som BHD og Coreg

opptil flere timer. Modellen består av konstruksjoner av grafer og beregninger av distanser i høyere dimensjoner for all data, samt gjennomgående vurderinger av store mengder av umarkerte data.
Dette fører til at modellen blir svært komplekse uten større fordeler i prediksjonsytelse, sammenliknet med veiledet maskinlæring med begrenset data. For konstruksjon av grafer krever BHD spesielt stor minneplass.
Dermed vil Alternativ 2 gis det score 2.

Alternativ 3 kan implementeres uten integrering av spektroskopisk sensor. Dette alternativet vil ha noen av de samme fordelene som alternativ 1, der det er tilgjengelig med etablerte modeller innen veiledet læring for implementering av en soft-sensor, i tillegg til å kunne benytte seg av semi-veiledede modeller etter behov, dersom det passer seg. Videre er den spektroskopiske sensoren ikke en fast måler i prosessen, dermed er dette alternativet uavhengig av en ekstern sensor. Dette alternativet er mer fleksibel, i tillegg til at datamengden som behandles er mindre.
Dermed tildeles det en score på 3.

Verdiskapning:

Siden dagens prosess ikke involverer noe form for vurderinger av produktkvalitet underveis i prosessen gis det en score 1.

Veiledede modeller presterer godt på tvers av alle kvalitetsmålinger. De har gode mulighet til å predikere produktkvalitet i større grad. Dette bidrar til at beslutninger som tas med hensyn til prediksjoner fra veiledede modeller kan være presise nok til å utgjøre en forskjell i prosessen. Imidlertid er det krevende å angi en terskel på hva som kan ansees som akseptable feilmarginer for prediksjoner av produktkvalitet. Av den grunn gis det en score på 3.

Det ble funnet ut at semi-veiledede modeller presterer dårligere i forhold til veiledede modeller. Dermed gis det en score på 2.

Alternativ 3 gir ingen god indikasjon på kvalitetsmålinger. Denne teknikken blir utkonkurrert mot standard statistiske estimater. Dersom denne predikerer feil, kan det føre til at nye beslutninger blir tatt på feil grunn og kan dermed føre til store tap av både tid og ressurser. Dermed blir det tildelt en score på 1.

Investering og vedlikehold:

Dagens prosess, ikke krever investering av verken spektroskopisk sensor eller innsamling av kvalitetsmålinger for modelleringsformål gis alternativet en score på 5.

Alternativet 1 er avhengig av både investering i spektroskopisk sensor og større mengder av kvalitetsmålinger for å operere. Når det er mindre mengder av kvalitetsmålinger å trene på viser det til større usikkerhet og svakere prediksjonsnøyaktighet rundt ytelsen til modellene. Dette blir presentert i Figur 5.8. Her er både standardavviket og RMSE-verdien høy for lavere treningsandeler av markerte data. Her er RFR et tilfelle som er

utsatt for dette problemet.
Alternativ 2 får en samlet score på: 2

Til tross for at semi-veiledede modeller benytter seg av større mengder med data, viser resultatene til at prediksjonsytelsen er lavere sammenliknet med veiledede modeller. Selv om, modellene er mer robust når mer data brukes, kan man ikke med sikkerhet si at prediksjonsnøyaktigheten forbedret i like stor skala. Som illustrert i Figur 5.7, vises det at RMSE-verdien til modellene ikke har en forventet nedgang i med økt mengde data. Denne varianten av soft-sensor har ikke gode nok resultater som veier opp for det reduserte kravet og mulig reduksjon av innsamling på kvalitetsmålinger.
Alternativ 2 får en samlet score på: 3

Alternativ 3 har krav på innsamling av kvalitetsmålinger for modelleringsformål, men har ingen behov for at en spektroskopisk sensor implementeres i prosessen. Denne modellen må likevel vedlikeholdes i lengre sikt ved å kontinuerlig forbedre ytelsen basert på nye innsamlede data. Dette kan derimot føre til feilaktige beslutninger tatt på feil estimering av produktkvalitet. Implementering av en soft-sensor krever altså monitorering over tid som kan bli løpende kostnader.
Alternativ 3 vil dermed få en score på 3.

5.3.2 Rangering av alternativer

Tabell 5.17: Oversikt over score og sum for ulike alternativer i MCDA-analyse. cellene under hvert alternativ følger Formel 4.2

Alternativ:	**Kriterie 1**	**Kriterie 2**	**Kriterie 3**	**Kriterie 4**	**Totalsum:**
Dagens Prosess	$1 * 0,30$	$4 * 0,30$	$1 * 0,30$	$5 * 0,30$	2,75
Alternativ 1	$4 * 0,25$	$2 * 0,25$	$3 * 0,25$	$2 * 0,25$	2,80
Alternativ 2	$3 * 0,20$	$1 * 0,20$	$2 * 0,20$	$3 * 0,20$	2,30
Alternativ 3	$1 * 0,25$	$3 * 0,25$	$1 * 0,25$	$3 * 0,25$	2,00

Beregning av resultater for MCDA analyser re gjort i Tabell 5.17. Alle cellene under hvert "Kriterie X" er regnet ut ifra Formel 4.2. Totalsummen viser den endelige vurderingen for hvert alternativ. MCDA-analysen forteller oss at Alternativ 1 er foretrukket.

Kapittel 6

Diskusjon

Denne studien har hatt som formål å hovedsakelig undersøke hvordan semi-veiledede modeller og spektroskopisk data kan bidra til å predikere produktkvalitet i biokjemisk produksjon. Basert på undersøkelse var det ønskelig å vurdere om integrering av en soft-sensor basert på en semi-veildet modell og spektroskopisk sensor kunne bidratt til å forbedre beslutninger tatt på vegne av produktkvalitet. I den sammenhengen ble ulike semi-veiledede algoritmer evaluert og sammenlignet med veiledede algoritmer. De semi-veiledede algoritmene inkluderte var COREG, BHD og en selvtrent RFR, og de veiledede algoritmene inkluderte KNR, SVR og RFR

En grundig analyse ble gjennomført på et reelt datasett fra en bioprosess hos Bioco AS. Datasettet inneholdt tre kvalitetsmålinger: Mw, SmallMoleules og BrixAdjusted. Analysen bestod av undersøkelse av ulike forsøk, som omfattet forskjellige strategier for oppdeling av trenings- og testdata, ulike metoder for inkludering av forskjellige mengder umarkerte data, samt hvordan ulike mengder markerte data treningsdata under optimalisering av modeller.

I det påfølgende kapittelet vil det bli diskutert tema som datasetttes innhold, oppgavens metodevalg og resultater, samt eventuelle utfordringer knyttet til oppgaven.

6.1 Konsekvenser av databehandling

6.1.1 Behandling av manglende verdier

For å kunne anvende en maskinlæringsmodell, er det et behov for både kvalitet og kvantitet i et datasett. Innholdet i datasettet, avhenger av hva informasjonen består av og hvordan den er hentet. Strukturen til datasettet setter grunnlaget for modelleringen av algoritmene. Når det er mangler på eller inkonsekvente verdier, kan det ha helhetlig en effekt som kan føre til at algoritmer får et dårlig utgangspunkt for å lære fra. Datasettet fra Bioco hadde opprinnelig 43 251 rader, men ble redusert til 28 701 rader etter behandling av data som forklart i Seksjon 5.2. Hovedårsaken til denne reduksjonen var manglende verdier.

En konsekvens av eliminerte datapunkter, var at det reduserte kvalitetsmålingene fra hele datasettet. De eliminerte observasjonene manglet opprinnelig verdier i forklaringsvariablene. Det var hovedsakelig målingene fra NIR-sensoren som utgjorde mesteparten av de manglende verdiene i datasettet. I utgangspunktet sto det mellom to muligheter for å håndtere denne utfordringen. Det første alternativet gikk ut på å eliminere observasjoner med manglende NIR-målinger. Det andre var å fjerne alle variablene som hadde målinger fra NIR i datasettet. Dersom det sistnevnte alternativet hadde blitt utført, ville datasettet beholdt alle kvalitetsmålingene sine.

Til tross for dette, ble det fortsatt valgt å eliminere observasjonene i datasettet. Grunnen til dette er fordi Tabell 3.1, viser til at observasjonene fra uke 49 var den eneste uken som ikke hadde registrerte NIR-målinger fra prosessen. Videre indikerer tabellen, at uke 47 og 49 behandlet råmaterialet likt. Eliminering av observasjoner fra uke 49 førte til en reduksjon på 72 kvalitetsmålinger. Dette var en betydelig reduksjon av responsverdiene. Elimineringen ble fortsatt utført, med tanke på at responsverdiene fra uke 47 potensielt kunne inneholde mye av informasjonen som var i den eliminerte andelen.
Figur 5.8 og Figur 5.10 viser blant annet ulike andeler av av markerte data som blir tilført i optimaliseringen flere maskinlæringsalgorimter. Det blir vist en trend der inkludering av flere kvalitetsmålinger i treningen fører til en forbedret prediksjonsytelse og robusthet hos de fleste modellene. Dette blir vist ved at RMSE-verdien blir lavere, samtidig som markerte andeler av data økte. En reduksjon på kvalitetsmålingene begrenser muligheten for at modellene predikerer optimalt. På grunn av denne årsakssammenhengen er det ønskelig med større eksponering av kvalitetsmålinger, for en forbedret prediksjonsytelse.

Konsekvenser av syntetisk data

Datasettet besto av betydelige og sammenhengende deler av data som hadde manglende verdier. Dette førte til at inkludering av syntetisk data ikke hadde hatt en positiv innvirkning på modellen. Selv om det er flere teknikker innen generering av syntetisk data er teknikken som oftest avhengig av at eksisterende data er konsise og representerer observasjonene godt. Dersom syntetisk data hadde blitt inkludert, til tross for de store manglene i datasettet, kunne genereringen ha potensielt basert seg på feil grunnlag. Tilføring av irrelevant data i datasettet kunne videre ha ført til å forvirre modellen og gitt en negativ påvirkning i prediksjonsytelsen. Dette var en av grunnene til at det heller ble besluttet å eliminere observasjoner med manglede verdier. Generelt sett kan reduksjon av data føre

til å svekke nøyaktigheten til modellen, likevel ble eliminering utført for å ikke svekke modellkvaliteten.

Vanligvis er det mer fordelaktig å generere syntetisk data istedenfor å eliminere observasjoner, gitt at datasettet hadde vært i en bedre tilstand. Grunnen til dette er for å beholde mest mulig av reell data. Til tross for det suboptimale datasettet, ble syntetisk data brukt for å tilføre ny data på noen enkelte observasjoner med manglende verdier i datasettet. Observasjonene med små intervaller av manglede verdier fikk ny syntetisk data, ved å bruke LOF. Imidlertid var det ikke mulig å gjøre dette for observasjonene som manglet NIR-målinger, fordi intervallet var stort. Hvis syntetisk data hadde erstattet disse verdiene, kunne det ha risikert å få nye problemer og redusert kvaliteten på datasettet drastisk. Videre hadde det introdusert utfordringer som skjevhet i datasettet. En fare med dette er at noen observasjoner kunne ha blitt over- eller underrepresentert. En konsekvens av dette er at modellene trener på feil informasjon.

Dersom disse problemene ikke hadde vært et tilfelle, hadde det vært mulig å unngå reduksjon i datasettet. Modelltreningen kunne ha inkludert mønstre fra datasettet som opprinnelig ikke ble brukt i predikeringsfasen i dette forsøket. Dersom det skulle vise seg å få noe skjevfordelt data, hadde det vært en mulighet å legge til ekstra trinn med krysssvalideringsteknikker for å monitorere treningsfasen, og undersøke om modellprestasjonen forverres.

6.2 Nedskalering av datasett som tidsseriedata

Det var besluttet å ikke behandlet datasettet videre etter inspeksjon for ekstremverdier. En videre behandling hadde vært nedskalering. Ved nedskalering ville antall observasjoner vært redusert med en angitt faktor for tidsintervall. En slik nedskalering hadde vært fordelaktig for reduksjon i prosesseringstid for de semi-veilede modellene. Samtidig kunne det ha økt kvalitet på de umarkerte observasjonene, da hver observasjon hadde bestått av mer unik informasjon. Uten nedjustering var det fare for at nærliggende observasjoner var bestående av lignende informasjon. Dette kunne ført til at de umarkerte observasjonene som ble benyttet av modellene, inneholdt lite variasjon. Med mindre variasjon, vil den informasjon som de utvalgte umarkerte observasjonene representerer, være informasjon som får større dominans i modelleringen. Når hele umarkerte datasettet benyttes, vil den overrepresenterte informasjon dominere i større grad i forhold til det underrepresenterte informasjon i de markerte observasjonene.

En betydelig årsak til prestasjonsytelsen til semi-veiledede modeller, er mengden informasjon i den umarkerte andelen av datasettet. Innholdet i datasettet var dominert av informasjon fra umarkerte observasjoner. En mulig løsning til å redusere kvantitet og øke kvalitet på informasjonen fra umarkerte observasjoner, hadde vært nedskalering. Ved nedskalering kunne informasjon mellom lignende observasjoner vært samlet og representert av færre observasjoner. På den måten ville antall umarkerte observasjoner vært redusert, samtidig som informasjon vært representert. Prosesseringstiden hadde vært betydelig redusert. Med reduksjon i antall observasjoner, ville det muligens vært mulig å redusere innvirkningen av umarkerte observasjoner.

Ved nedskalering hadde alle observasjoner måtte blitt behandlet likt. Ved en slik behandling, ville forholdet mellom markerte og umarkerte observasjoner forblitt det samme. Av den grunn vil det ikke vært en reduksjon i innvirkning fra de umarkerte observasjonene. Men en nedskaleringsmetode med der informasjonen i observasjonene skal bevares til best grad, hadde det vært muligheter for at informasjon fra umarkerte hadde påvirket informasjon i det markerte. Dette vil svekket undersøkelsen om hvordan umarkerte observasjoner tilfører supplerende informasjon som modeller kan benytte. Dersom de markerte observasjonene blir behandlet slik at informasjonen fra nærliggende og umarkerte observasjoner blir representert i dem i ettertid, ville en slik analyse vært forsvarlig. Skille mellom markerte og umarkerte data vil ikke være like tydelig.

En fordel med nedskaleringen, hadde vært reduksjon i antall observasjoner med enzymtype "A2" fra normalproduksjonen. Spesielt ved standardisering, vil observasjoner med denne enzymtypen bli vektet høyere enn andre. Når visse kvalitetsmålinger som "Mw" er sterkt knyttet til enzymtype, har en mer dominere enzymtype hatt en negativ innvirkning på modelleringen.

Ved tilfeller når enzymtyper endrer seg under et bestemt tidsintervall, ville informasjonen mellom enzymtyper blandet seg. Da ville informasjon ikke lenger være knyttet til hver spesifikk enzymtype, som hadde redusert tolkbarheten av resultatene.

På en annen side var observasjonene justert for å redusere støy med et medianfilter på 30 minutter. Av den grunn er det allerede noe informasjon som har blitt blandet mellom markerte og umarkerte observasjoner.

6.3 Forholdet mellom markerte og umarkerte data

Et formål med oppgaven var å utforske effekten av hvordan informasjonen i markerte data og umarkerte data kan påvirke ytelsen til semi-veiledede modeller. Forskningen skulle sammenlignes med veiledede modeller, som benytter seg av markerte data. Siden semi-veiledede modeller har tilgang til mer data, kan det føre til at slike modeller oppgaver visse mønstre og sammenhenger som ikke er tilstedet i de markerte data. Dette er imidlertid avhengig av i hvor stor grad informasjonen i de umarkerte data avviker fra informasjonen som forventes i ny og usett data.

Datasettet fra Bioco AS består av markerte og umarkerte data som inneholder ulike fordelinger av informasjon. Som vist i Seksjon 5.1, viser Figur 5.1 et eksempel på hvordan fordelingen av informasjon skiller seg mellom markerte og umarkerte data i datasettet. Figur 5.2 viser hvordan enzymtyper er fordelt i markerte og umarkerte data. Siden standardproduksjon og annen testproduksjon er forbundet med enzymtyper, vil en kombinasjon av figurene vise til hvordan informasjonen i markerte og umarkerte data skiller seg fra hverandre i datasettet.

Evalueringen av hvordan den informasjonen i umarkerte data påvirker prediksjonsytelse til modelllen, er hovedsakelig avhengig av hvordan testdata utformes. Siden testdata

trenger faktiske verdier i responsen som kan brukes til å evaluere predikerte verdier, vil det utformes fra markerte data. Av den grunn vil informasjonen i testdata basere seg kun på markerte data.

Veiledede modeller evalueres på testsettet som består av markerte data, og vil dermed ha en fordel. Siden både trenings- og testdata kommer fra samme del av datasettet, vil veiledede modeller muligens kunne lære mønstre og sammenhenger som er representert i testsettet. Det kan være flere faktorer som struktur og parametere til en algoritme som kan være årsaken til forbedret ytelse i forhold til semi-veiledede modeller. Imidlertid er flere av de valgte semi-veiledede modeller basert på lignende strutkur og parametre. Til tross for dette, presterte veiledede modeller generelt sett bedre over alle undersøkelser var gjennomført i studien. Siden hovedvekten av informasjonen i markerte data kommer fra designproduksjon, vil veiledede modeller ha fordel ved å ha den informasjonen vektet under treningsfasen.

Semi-veiledede modeller utnytter derimot umarkerte data som hovedsakelig representerer spesifikasjoner fra standardproduksjon. Dette vil dermed i utgangpunktet være en ulempe når informasjonen i testsettet er vektet ulikt. Ulik informasjon kan gi modellen innsikt i nye mønstre som kan forbedre prediksjonsytelsen. For stor mengde av ulik informasjon kan imidlertid gi feilaktige sammenhenger. Det kan påvirke prediksjonsytelse til modellen negativt. Når innholdet i markerte og umarkerte data ligner på hverandre, kan det føre til overtilpasning.

Undersøkelser på hvordan innholdet i umarkerte data kan påvirke modellen ble utforsket gjennom ulike strategier for uthenting av umarkerte data. For oppdelingsstrategi Alternativ 1: Enzymtype, var umarkerte data hentet slik at informasjon var lignende det som var i de markerte data. Fra resultatene, spesielt Tabell 5.2 og 5.3, viser at de best presterende semi-veiledede modeller benyttet seg av den største andelen av umarkerte data som selektivt var utvalgt, framfor alle de umarkerte data. Det er kan ses ut ifra antall '*' ved modellnavnene. 5.7 viser også en positiv utvikling for gjennomsnitt RMSE-score fram til alle umarkerte data tildeles modellene.

Alternativ 2: Dag og kontinuitet er en oppdelingsstrategi som viser hvordan semi-veiledede modeller handterer data som er ulik de markerte data. I dette tilfelle viser Tabell 5.10 og Tabell 5.11 viser at semi-veiledede modeller foretrekker egen vurdering av de umarkerte framfor de selektivt valgte delene av de umarkerte data.

Imidlertid er dette en konsekvens av utforming av testsettet. Siden testsettet er basert på innholdet i det markerte data, vil optimalering etter et valideringsettet som også inneholder dette, foretrekker parametre og umarkerte data som innholder de samme mønstre. Et annet testsett som hadde representert en vektet tilnærming til begge typer informasjon, både standardproduksjon og designproduksjon, kunne gitt andre resultater og en balansert tilnærming til vurdering av både veiledede og semi-veiledede modeller.

Kapittel 7

Konklusjon

Dette studie har forsøkt å svare på følgende problemstilling:

Hvordan kan regresjons-algoritmer basert på semi-veiledet læring, bidra til å forbedre prediksjon av produktkvalitet i biokjemisk prosessindustrielle data sammenlignet med veiledet læring, ved å utnytte umarkerte data og spektroskopisk måleteknologi?

Problemstillingen ble brutt ned til fire forskningsspørsmål som skulle bidra til å avdekke relevant data og innsikt til å forstå hvordan problemstillingen kunne besvares. Følgende spørsmål er:

Forskningsspørsmål 1: *I hvilken grad skiller prediksjonsytelsen av semi-veiledede modeller seg fra klassiske, veiledede modeller?*

Forskningsspørsmål 2: *Hvilke muligheter og utfordringer er knyttet til informasjon i umarkerte data som påvirker ytelsen til semi-veiledede modeller?*

Forskningsspørsmål 3: *Hvordan endres prediksjonsnøyaktigheten på produktkvalitet ved å benytte data av spektroskopiske målinger?*

Forskningsspørsmål 4: *I hvor stor grad kan prediksjonsnøyaktigheten til maskinlæringsalgoritmer med tilgang på spektroskopisk informasjon, bidra til økt verdiskapning?*

For å besvare på forskningsspørsmålene i denne oppgaven, ble det benyttet totalt 6 maskinlæringsalgoritmer. Forsøket gikk ut på å undersøke 3 semi-veiledede og 3 veiledede modeller på et datasett fra Bioco. Det opprinnelige datasettet som besto av 3 ulike responsvariabler, ble 3 separate datasett med en responsvariabel hver. Algoritmene ble modellert på disse. Videre ble det undersøkt om data fra en spektroskopisk sensor hadde bidratt til å forbedre prediksjon av maskinlæringsalgoritmer. I tillegg til ble det utført forsøk på hvordan modellene presterte med eksponering av ulike mengder markerte og umarkerte data.

Resultatene fra studien viser til at veiledede algoritmer presterer bedre enn semi-veiledede algoritmer. Dette kan være på grunn av at markerte data skiller seg fra umarkerte data fra datasettet. Semi-veiledede modeller kan lære informasjon som påvirker deres prestasjonsytelse negativ. For inkludering av spektroskopisk data viser til betydelig forbedring

i prediksjonsytelse til maskinlæringsmodeller. Imidlertid er det krevende å avgjøre hvor stor grad av forbedring som rettferdiggjør investering av en spektroskopisk sensor.

I lys av våre funn rundt forskningsspørsmålene ble det funnet ut at i noen tilfeller presterer semi-veiledet læring bra, men helhetlig vil veiledede modeller prestere bedre. Datasettet som ble brukt i forsøket er unikt. Siden de teoretiske fordelene til semi-veiledede modeller ikke ble realisert i oppgaven, ble det funnet ut at fordelene med disse algoritmene ikke veiet opp for å redusere kravet for kvalitetsmålinger.

MCDA analysen konkluderer med at alternativ 1, veiledet maskinlæring med spektroskopisk data, er den foretrekkende alternative løsningen, som bør implementeres. Bioco begrenset informasjon om de økonomiske sidene om produktinformasjon, vedlikehold og investeringsinformasjon. Dette fører til at anbefalingen er lite holdbar.

Det konkluderes med kun tre spesifikke algoritmer innen semi-veiledet læring ble benyttet og dermed er alle vurderinger tatt i oppgaven, begrenset til disse algoritmene. De representerer ikke veiledede modeller.

Blank Side

Bibliografi

[1] Tilman Klaeger, Sebastian Gottschall og Lukas Oehm. "Data Science on Industrial Data - Today's Challenges in Brown Field Applications". I: *Challenges* 12 (1 jan. 2021), s. 2. DOI: 10.3390/challe12010002.

[2] R Crawshaw. "Food Industry Co-Products as Animal Feeds". I: *Handbook of Waste Management and Co-Product Recovery in Food Processing*. Elsevier, 2009, s. 391–411.

[3] Sara N Garcia, Bennie I Osburn og Michele T Jay-Russell. "One health for food safety, food security, and sustainable food production". I: *Frontiers in Sustainable Food Systems* 4 (2020), s. 1.

[4] Yafeng Guo og Biao Huang. "State Estimation Incorporating Infrequent, Delayed and Integral Measurements". I: *Automatica* 58 (2015), s. 32–38. DOI: 10.1016/j.automatica.2015.05.001.

[5] Temilade Abass mfl. "Concept paper: Innovative Approaches to Food Quality Control: AI and Machine Learning for Predictive Analysis". I: *World Journal of Advanced Research and Reviews* 21 (3 2024), s. 823–828. DOI: 10.30574/wjarr.2024.21.3.0719. URL: https://doi.org/10.30574/wjarr.2024.21.3.0719.

[6] Alberto Garre, Mari Carmen Ruiz og Eloy Hontoria. "Application of Machine Learning to Support Production Planning of a Food Industry in the Context of Waste Generation Under Uncertainty". I: *Operations Research Perspectives* 7 (2020), s. 100–147. DOI: 10.1016/j.orp.2020.100147. URL: www.elsevier.com/locate/orp.

[7] Mohd Javaid mfl. "Significance of sensors for industry 4.0: Roles, capabilities, and applications". I: *Sensors International* 2 (2021), s. 100110.

[8] Jan U Porep, Dietmar R Kammerer og Reinhold Carle. "On-Line Application of Near Infrared (NIR) Spectroscopy in Food Production". I: *Trends in Food Science & Technology* 46 (2 2015), s. 211–230. DOI: 10.1016/j.tifs.2015.10.002. URL: http://dx.doi.org/10.1016/j.tifs.2015.10.002.

[9] Jens Claßen mfl. "Spectroscopic sensors for in-line bioprocess monitoring in research and pharmaceutical industrial application". I: *Analytical and bioanalytical chemistry* 409 (2017), s. 651–666.

[10] Petr Kadlec, Bogdan Gabrys og Sibylle Strandt. "Data-Driven Soft Sensors in the Process Industry". I: *Computers and Chemical Engineering* 33 (2009), s. 795–814. DOI: 10.1016/j.compchemeng.2008.12.012.

[11] Oliver J Fisher mfl. "Considerations, challenges and opportunities when developing data-driven models for process manufacturing systems". I: *Computers & Chemical Engineering* 140 (2020).

[12] Francisco A A Souza, Rui Araújo og Jérôme Mendes. "Review of Soft Sensor Methods for Regression Applications". I: (2015). DOI: 10.1016/j.chemolab.2015.12.011.

[13] Weiwu Yan, Di Tang og Yujun Lin. "A data-driven soft sensor modeling method based on deep learning and its application". I: *IEEE Transactions on Industrial Electronics* 64.5 (2016), s. 4237–4245.

[14] Najibesadat Sadati, Ratna Babu Chinnam og Milad Zafar Nezhad. "Observational data-driven modeling and optimization of manufacturing processes". I: *Expert Systems with Applications* 93 (2018), s. 456–464.

[15] Yu-Feng Li, Han-Wen Zha og Zhi-Hua Zhou. "Learning safe prediction for semi-supervised regression". I: *Proceedings of the AAAI Conference on Artificial Intelligence*. Bd. 31. 1. 2017.

[16] Zhi-Hua Zhou og Ming Li. "Semi-Supervised Regression with Co-Training". I: *IJCAI* 5 (2005), s. 908–913. URL: https://www.mit.bme.hu/eng/system/files/oktatas/targyak/7153/SemiSupervisedRegressionWithCoTraining_Zhou.pdf.

[17] Sebastian Raschka og Vahid Mirjalili. *Python Machine Learning: Machine Learning and Deep Learning with Python, Scikit-Learn, and TensorFlow 2, 3rd Edition*. 3. utg. Packt Publishing, sep. 2019. ISBN: 978-1787125933.

[18] Tomasz Burzykowski mfl. "Introduction to Machine Learning". I: *American Journal of Orthodontics and Dentofacial Orthopedics* 163 (2023), s. 732–734. DOI: 10.1016/j.ajodo.2023.02.005. URL: https://doi.org/10.1016/j.ajodo.2023.02.005.

[19] Allen H. Renear, Simone Sacchi og Karen M. Wickett. "Definitions of Dataset in the Scientific and Technical Literature". I: bd. 47. Nov. 2010. DOI: 10.1002/meet.14504701240.

[20] Donna L Mohr, William J Wilson og Rudolf J Freund. *Statistical methods*. Academic Press, 2021.

[21] Sandhya N. Dhage og Charanjeet Kaur Raina. "A Review on Machine Learning Techniques". I: *International Journal on Recent and Innovation Trends in Computing and Communication* 4 (3 2016), s. 395–399. ISSN: 2321-8169. URL: http://www.ijritcc.org.

[22] Shruthi H. Shetty mfl. "Supervised Machine Learning: Algorithms and Applications". I: *Department of ECE, Sahyadri College of Engineering & Management* (2022), s. 4–5. DOI: 10.1002/9781119821908.ch1. URL: https://www.researchgate.net/publication/358216497.

[23] Zoubin Ghahramani. "Unsupervised Learning". I: (2004), s. 72–112. URL: http://www.gatsby.ucl.ac.uk/~zoubin.

[24] Svante Wold, Kim Esbensen og Paul Geladi. "Principal Component Analysis". I: *Chemometrics and intelligent laboratory systems* 2.3 (1987), s. 37–52.

[25] Hervé Abdi og Lynne J Williams. "Principal Component Analysis". I: *Wiley interdisciplinary reviews: computational statistics* 2.4 (2010), s. 433–459.

[26] Yu-Feng Li og De-Ming Liang. "Safe Semi-Supervised Learning: A Brief Introduction". I: *Frontiers of Computer Science* 13 (4 2019), s. 669–676. DOI: 10.1007/s11704-019-8452-2. URL: https://doi.org/10.1007/s11704-019-8452-2.

[27] Kevin P. Murphy. *Probabilistic Machine Learning: An Introduction*. Red. av Francis Bach. 1. utg. Bd. 1. MIT Press, mar. 2022, s. 633–643.

[28] Olivier Chapelle, Bernhard Schölkopf og Alexander Zien. *Semi-Supervised Learning*. Bd. 20. MIT Press, mar. 2010. ISBN: 9780262514125. URL: https://books.google.no/books?id=zHAOQgAACAAJ.

[29] Jesper E. van Engelen og Holger H. Hoos. "A Survey on Semi-Supervised Learning". I: *Machine Learning* 109 (2 feb. 2020), s. 373–440. ISSN: 15730565. DOI: 10.1007/s10994-019-05855-6.

[30] Avrim Blum og Tom Mitchell. "Combining Labeled and Unlabeled Data with Co-Training". I: (1998), s. 92–100.

[31] Xiaojin Zhu og Zoubin Ghahramani. "Learning from Labeled and Unlabeled Data with Label Propagation". I: *ProQuest number: information to all users* (2002).

[32] Pauline Wauquier og Mikaela Keller. "A Metric Learning Approach for Graph-Based Label Propagation". I: *arXiv preprint arXiv:1511.05789* (2015).

[33] Tomas Borovicka mfl. "Selecting Representative Data Sets". I: *Advances in data mining knowledge discovery and applications* 12 (2012), s. 43–70. DOI: 10.5772/50787. URL: http://dx.doi.org/10.5772/50787.

[34] Batta Mahesh. "Machine learning Algorithms-a Review". I: *International Journal of Science and Research (IJSR).[Internet]* 9.1 (2020), s. 381–386.

[35] Dar Masroof Amin og Munishwar Rai. "A Clustering Hybrid Algorithm for Smart Datasets Using Machine Learning". I: *International Journal of Advanced Computer Science and Applications* 11 (9 2020). URL: www.ijacsa.thesai.org.

[36] Xue Ying. "An Overview of Overfitting and Its Solutions". I: *Journal of Physics: Conference Series*. Bd. 1168. IOP Publishing. 2019, s. 022022.

[37] Zheng Xiong mfl. "Evaluating Explorative Prediction Power of Machine Learning Algorithms for Materials Discovery Using kKfold Forward Cross-validation". I: *Computational Materials Science* 171 (2020), s. 109203.

[38] Ron Kohavi. "A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection". I: 14 (2 1995). URL: http://robotics.stanford.edu/~ronnyk.

[39] Bruce G. Marcot og Anca M. Hanea. "What Is an Optimal Value of K in K-fold Cross-validation in Discrete Bayesian Network Analysis?" I: *Computational Statistics* 36 (3 sep. 2021), s. 2009–2031. ISSN: 16139658. DOI: 10.1007/s00180-020-00999-9.

[40] Sashikanta Prusty, Srikanta Patnaik og Sujit Kumar Dash. "SKCV: Stratified K-fold Cross-validation on ML Classifiers for Predicting Cervical Cancer". I: *Frontiers in Nanotechnology* 4 (2022), s. 972421.

[41] Ji-Hyun Kim. "Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap". I: *Computational statistics & data analysis* 53.11 (2009), s. 3735–3745.

[42] Mahan Hosseini mfl. “I Tried a Bunch of Things: The Dangers of Unexpected Overfitting in Classification of Brain Data”. I: *Neuroscience & Biobehavioral Reviews* 119 (2020), s. 456–467.

[43] Alexei Botchkarev. “Performance Metrics (Error Measures) in Machine Learning Regression, Forecasting and Prognostics: Properties and Typology”. I: *arXiv preprint arXiv:1809.03006* (2018).

[44] Tianfeng Chai, Roland R Draxler mfl. “Root Mean Square Error (RMSE) or Mean Absolute Eerror (MAE)”. I: *Geoscientific model development discussions* 7.1 (2014), s. 1525–1534.

[45] Timothy O Hodson. “Root Mean Square Error (RMSE) or Mean Absolute Error (MAE): When to Use Them or Not”. I: *Geoscientific Model Development Discussions* 2022 (2022), s. 1–10.

[46] Ferenc Moksony og Rita Heged. “Small Is Beautiful. The Use and Interpretation of R2 in Social Research”. I: *Szociológiai Szemle, Special issue* (1990), s. 130–138.

[47] Inge S Helland. “On the Interpretation and Use of R2 in Regression Analysis”. I: *Biometrics* (1987), s. 61–69.

[48] Cort J Willmott og Kenji Matsuura. “Advantages of the Mean Absolute Error (MAE) over the Root Mean Square Error (RMSE) in Assessing Average Model Performance”. I: *Climate research* 30.1 (2005), s. 79–82.

[49] Davide Chicco, Matthijs J Warrens og Giuseppe Jurman. “The Coefficient of Determination R-squared Is More Informative Than SMAPE, MAE, MAPE, MSE and RMSE in Regression Analysis Evaluation”. I: *Peerj computer science* 7 (2021), e623.

[50] Arnaud De Myttenaere mfl. “Mean Absolute Percentage Error for Regression Models”. I: *Neurocomputing* 192 (2016), s. 38–48.

[51] Tong Yu og Hong Zhu. “Hyper-parameter Optimization: A Review of Algorithms and Applications”. I: *arXiv preprint arXiv:2003.05689* (2020).

[52] James Bergstra mfl. “Algorithms for Hyper-parameter Optimization”. I: *Advances in neural information processing systems* 24 (2011).

[53] Sayan Putatunda og Kiran Rama. “A Comparative Analysis of Hyperopt as Against Other Approaches for Hyper-parameter Optimization of XGBoost”. I: *Proceedings of the 2018 International Conference on Signal Processing and Machine Learning*. 2018, s. 6–10.

[54] John M Lachin. “Fallacies of last observation carried forward analyses”. I: *Clinical trials* 13.2 (2016), s. 161–168.

[55] Muhammad Umer Farooq og Aemal Khattak. “Exploring Statistical and Machine Learning-Based Missing Data Imputation Methods to Improve Crash Frequency Prediction Models for Highway-Rail Grade Crossings”. I: *International Road Federation (IRF) Global R2T Conference & Exhibition* (2023), s. 1–14.

[56] G.E.P. Box mfl. *Time Series Analysis: Forecasting and Control*. Wiley Series in Probability and Statistics. Wiley, 2015. ISBN: 9781118674925. URL: `https://books.google.no/books?id=rNt5CgAAQBAJ`.

[57] Pınar Ersoy. “Evolution of Outlier Algorithms for Anomaly Detection”. I: *Manchester Journal of Artificial Intelligence and Applied Sciences* 2.1 (2021).

[58] Geoff Cumming og Robert Calin-Jageman. *Introduction to the new statistics: Estimation, open science, and beyond*. Routledge, 2016.

[59] Kejin Hu. "Become Competent Within One Day in Generating Boxplots and Violin Plots for a Novice Without Prior R Experience". I: *Methods and protocols* 3.4 (2020), s. 64.

[60] Michael C Thrun, Tino Gehlert og Alfred Ultsch. "Analyzing the Fine Structure of Distributions". I: *PloS one* 15.10 (2020), e0238835.

[61] Philip Sedgwick. "Pearson's Correlation Coefficient". I: *Bmj* 345 (2012).

[62] Douglas C Montgomery, Elizabeth A Peck og G Geoffrey Vining. *Introduction to linear regression analysis*. John Wiley & Sons, 2021.

[63] Chih-Jen Lin, Ruby C Weng mfl. "Simple probabilistic predictions for support vector regression". I: *National Taiwan University, Taipei* (2004).

[64] J R Quinlan. "Induction of Decision Trees". I: *Machine Learning* 1 (1986), s. 81–106.

[65] Matt Gifford og Tuncay Bayrak. "A Predictive Analytics Model for Forecasting Outcomes in the National Football League Games Using Decision Tree and Logistic Regression". I: *Decision Analytics Journal* 8 (sep. 2023), s. 100–296. ISSN: 27726622. DOI: 10.1016/j.dajour.2023.100296.

[66] Anuradha Chokka og K. Sandhya Rani. "PCA Based Regression Decision Tree Classification for Somatic Mutations". I: *International Journal of Engineering and Advanced Technology* 8 (6 sep. 2019), s. 1095–1102. ISSN: 22498958. DOI: 10.35940/ijeat.F1181.0986S319.

[67] Victor Rodriguez-Galiano mfl. "Machine Learning Predictive Models for Mineral Prospectivity: An Evaluation of Neural Networks, Random Forest, Regression Trees and Support Vector Machines". I: *Ore Geology Reviews* 71 (2015), s. 804–818. DOI: 10.1016/j.oregeorev.2015.01.001. URL: http://dx.doi.org/10.1016/j.oregeorev.2015.01.001.

[68] Ye Ren mfl. "Ensemble Classification and Regression-Recent Developments, Applications and Future Directions". I: *IEEE Computational intelligence magazine* 11 (1 2016), s. 41–53. DOI: 10.1109/MCI.2015.2471235. URL: https://www.researchgate.net/publication/290476291.

[69] V. Kishore Ayyadevara. "Gradient Boosting Machine". I: *Pro Machine Learning Algorithms : A Hands-On Approach to Implementing Algorithms in Python and R*. Berkeley, CA: Apress, 2018, s. 117–134. ISBN: 978-1-4842-3564-5. DOI: 10.1007/978-1-4842-3564-5_6. URL: https://doi.org/10.1007/978-1-4842-3564-5_6.

[70] Cao Truong Tran mfl. "Bagging and Feature Selection for Classification with Incomplete Data". I: (2017), s. 471–486.

[71] Yousef Elgimati. "Weighted Bagging in Decision Trees: Data Mining". I: *JINAV: Journal of Information and Visualization* 1.1 (2020), s. 1–14.

[72] JiSoo Ham mfl. "Investigation of the Random Forest Framework for Classification of Hyperspectral Data". I: *IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING* 43 (3 2005), s. 492–501. DOI: 10.1109/TGRS.2004.842481.

[73] Chao Hu mfl. "Data-driven Method Based on Particle Swarm Optimization and K-nearest Neighbor Regression for Rstimating Capacity of Lithium-ion Battery". I: *Applied Energy* 129 (2014), s. 49–55.

[74] Mohan Timilsina, Alejandro Figueroa og Haixuan Yang. "Semi-Supervised Regression Using Diffusion on Graphs". I: *Applied Soft Computing Journal* 104 (2021), s. 107–188. DOI: 10.1016/j.asoc.2021.107188. URL: http://creativecommons.org/licenses/by/4.0/.

[75] Pete Chapman mfl. "Crisp-Dm 1.0. Step-by-Step Data Mining Guide". I: *CRISP-DM Consortium* (2000). ISSN: 0957-4174. DOI: 10.1109/ICETET.2008.239.

[76] Asbjørn Rolstadås mfl. *Praktisk Prosjektledelse: fra Idé til Gevinst*. 2. utg. Fagbokforlaget, 2020.

[77] C Michael mfl. "Structured Decision Making: Case Studies in Natural Resource Management". I: *The Journal of Wildlife Management* 85 (6 2021), s. 52–53. DOI: 10.1002/jwmg.22050.

[78] Tone Aspevik mfl. "Valorization of Proteins from Co- and By-Products from the Fish and Meat Industry". I: *Topics in Current Chemistry* 375 (). DOI: 10.1007/s41061-017-0143-6.

[79] Leandro Soares Oliveira, Leandro S Oliveira og Adriana S Franca. "Applications of Near Infrared Spectroscopy (NIRS) in Food Quality Evaluation". I: *Food Quality: Control, Analysis and Consumer Concerns* 4 (3 2011), s. 131–179. URL: https://www.researchgate.net/publication/287243398.

[80] Indurani Chandrasekaran mfl. "Potential of Near-Infrared (NIR) Spectroscopy and Hyperspectral Imaging for Quality and Safety Assessment of Fruits: An Overview". I: *Food Analytical Methods* 12 (2019), s. 2438–2458. DOI: 10.1007/s12161-019-01609-1. URL: https://doi.org/10.1007/s12161-019-01609-1.

[81] Bioco. *Bioco - Om oss*. 2024. URL: https://www.bioco.no/om-oss (sjekket 10.06.2024).

[82] Nofima. *Nofima - Om oss*. 2024. URL: https://nofima.no/om-oss/ (sjekket 23.05.2024).

[83] Digifoods. *Digifoods - About*. 2024. URL: https://digifoods.no/about/ (sjekket 10.06.2024).

[84] Hyndman Rob J og Athanasopoulos Georg. *Forecasting: Principles and Practice*. 3. utg. OTexts, 2021. URL: https://otexts.com/fpp3/index.html.

[85] Harry J. Foxwell. *Creating Good Data: A Guide to Dataset Structure and Data Representation*. Apress Media LLC, jan. 2020, s. 1–105. ISBN: 9781484261033. DOI: 10.1007/978-1-4842-6103-3.

[86] Matthew Ryan Lavery mfl. "Number of Predictors and Multicollinearity: What Are Their Effects on Error and Bias in Regression?" I: *Communications in Statistics-Simulation and Computation* 48 (1 2019), s. 27–38. ISSN: 1532-4141. DOI: 10.1080/03610918.2017.1371750. URL: https://www.tandfonline.com/action/journalInformation?journalCode=lssp20.

[87] Roy Jafari. "Hands-On Data Preprocessing in Python: Learn How to Effectively Prepare Data for Successful Data Analytics". I: (2022).

[88] Tlamelo Emmanuel mfl. "A Survey on Missing Data in Machine Learning". I: *Journal of Big Data* (2021). DOI: 10.1186/s40537-021-00516-9. URL: https://doi.org/10.1186/s40537-021-00516-9.

[89] Shahidul Islam Khan og Abu Sayed Md Latiful Hoque. "SICE: An Improved Missing Data Imputation Technique Background and Related Works". I: 7 (2020), s. 37. DOI: 10.1186/s40537-020-00313-w. URL: http://creativecommons.org/licenses/by/4.0/..

[90] Soledad Galli. *Python Feature Engineering Cookbook*. 2. utg. Packt Publishing Limited, 2022, s. 33–34. ISBN: 9781804611302.

[91] Tanay Agrawal. *Hyperparameter Optimization in Machine Learning: Make Your Machine Learning and Deep Learning Models More Efficient*. Apress Media LLC, jan. 2020, s. 1–166. ISBN: 9781484265796. DOI: 10.1007/978-1-4842-6579-6.

[92] Mohan Timilsina. *Semi-Supervised Regression*. 2019. URL: https://github.com/timilsinamohan/SSR?tab=readme-ov-file#readme.

[93] Fabian Pedregosa mfl. "Scikit-learn: Machine Learning in Python Gaël Varoquaux Bertrand Thirion Vincent Dubourg Alexandre Passos PEDREGOSA, VAROQUAUX, GRAMFORT ET AL. Matthieu Perrot". I: *Journal of Machine Learning Research* 12 (2011), s. 2825–2830. URL: http://scikit-learn.sourceforge.net..

[94] Takuya Akiba mfl. "Optuna: A Next-generation Hyperparameter Optimization Framework". I: (2019). URL: https://github.com/pfnet/optuna/.

[95] Fabian Pedregosa mfl. "Scikit-learn: Machine Learning in Python Gaël Varoquaux Bertrand Thirion Vincent Dubourg Alexandre Passos PEDREGOSA, VAROQUAUX, GRAMFORT ET AL. Matthieu Perrot". I: *The Journal of Machine Learning Research* 12 (2011), s. 2825–2830. URL: http://scikit-learn.sourceforge.net..

[96] Lin-Han Jia mfl. "LAMDA-SSL: A Comprehensive Semi-Supervised Learning Toolkit". I: *arXiv preprint arXiv:2208.04610* (2022). URL: https://github.com/YGZWQZD/LAMDA-SSL..

Blank Side

Tillegg A

Vedlegg

A.1 Maskinvare og programvare

Følgende maskinvarer ble benyttet til oppgavens formål:

- Apple MacBook Pro 2021. M1 Pro chip. 16 GB Ram. Lagringsplass 512 GB. Operativsystem: macOS Sonoma 14.4.1.
- Apple MacBook Pro 2020. i7 Core. Intel Iris Plus Graphics 16 GB Ram. Lagringsplass 512 GB. Operativsystem: macOS Sonoma 14.4.1.
- Apple MacBook Pro 2023. M3 Max chip. 36 GB Ram. Lagringsplass 512 GB. Operativsystem: macOS Sonoma 14.4.1
- SuperPC. 2 x Intel Xeon Gold 6238R. 112 Kjerner. 2.2 GHz. 1.5 TB Ram. Operativsystem: Rocky Linux 8.9 (Green Obsidian) x86_64

Primært ble programmeringsverktøyet Google Colab benyttet til å skrive og lese programmeringsspråket Python, i versjon 3.10.12.

A.2 Versjonkontroll

GitHub ble benyttet som plattform der all kode i sammenheng med prosjektet ble gjort tilgjengelig. Bruk av versjonskontroll gir oversikt over endringer over tid og muligheten til å integrere disse på en strukturert måte. Det sikrer sporbarhet og reduserer muligheten for tap av arbeid og data.

A.3 Benyttede bibliotek og pakker

A.3.1 Scikit-learn

Scikit-learn er et åpent kildekode bibliotek som benyttes for å anvende maskinlæringsalgoritmer i Python. Dette er et "programmeringsgrensesnitt" (*eng. Application Programming Interface (API)*) som tilbyr brukeren å hente ut algoritmer fra et bibliotek publisert i GitHub og importere det i et Python script. Dette biblioteket tilbyr støtte for både veiledede og ikke-veiledede (se Seksjon 2.1.2) maskinlæringsmodeller. Scikit-learn blir videre brukt for å blant annet preprosessering, rensing og visualisering av data. Til tross for at dette biblioteket er enkelt å bruke, viser en studie av Fabian Pedregosa i 2011 at andelen av testing som har blitt utført på koden i biblioteket er på omtrent 81 % [95]. Prosentandelen viser til at kvalitet er like høyt prioritert som brukervennlighet.

A.3.2 LAMDA

Lamda er et bibliotek i Python som inneholder totalt 30 maskinlæringsmodeller innen semi-veiledet læring (se Seksjon 2.1.2), der 12 av disse er statistiske maskinlæringsmodeller. Biblioteket er laget på grunnlag av kjente maskinlæringsalgoritmer fra bibliotek som Scikit-learn, med en lisens som gir tilgang til å bruke det slik en ønsker. Her finnes det algoritmer for klassifikasjon, kluster og regresjonsproblemer. Dette biblioteket inneholder kun en algoritme som løser regresjonsproblemer innen semi-veiledet læring, nemlig Coreg (se Seksjon 2.2.2).

Dokumentasjon og tilgang til Github for biblioteket LAMDA-SSL kan bli funnet her: [96].

A.3.3 Optuna

For justering og optimering av modellparamtere, har pakken Optuna blitt benyttet. Optuna er et optimeringsverktøy som brukes for å finne best mulig kombinasjon av hyperparametere til en maskinlæringsmodell. Dette er en åpen kildekode bibliotek i Python, som er laget for å automatisere den manuelle prosessen som velger optimale sett med hyperparametere. Optuna presterer godt ved å bruke færre iterasjoner sammenliknet med andre typiske hyperparameter optimeringsverktøy som rutenettsøk eller ved tilfeldig utplukking av parametere. I tillegg til å være en fleksibel teknikk, har den mulighet til å visualisere prosessen sin som hjelper brukeren med å forstå hvilke resultater som er relevante for å regne seg fram til optimale parameterverdier [94].

A.3.4 Benyttede versjoner av bibliotek og pakker

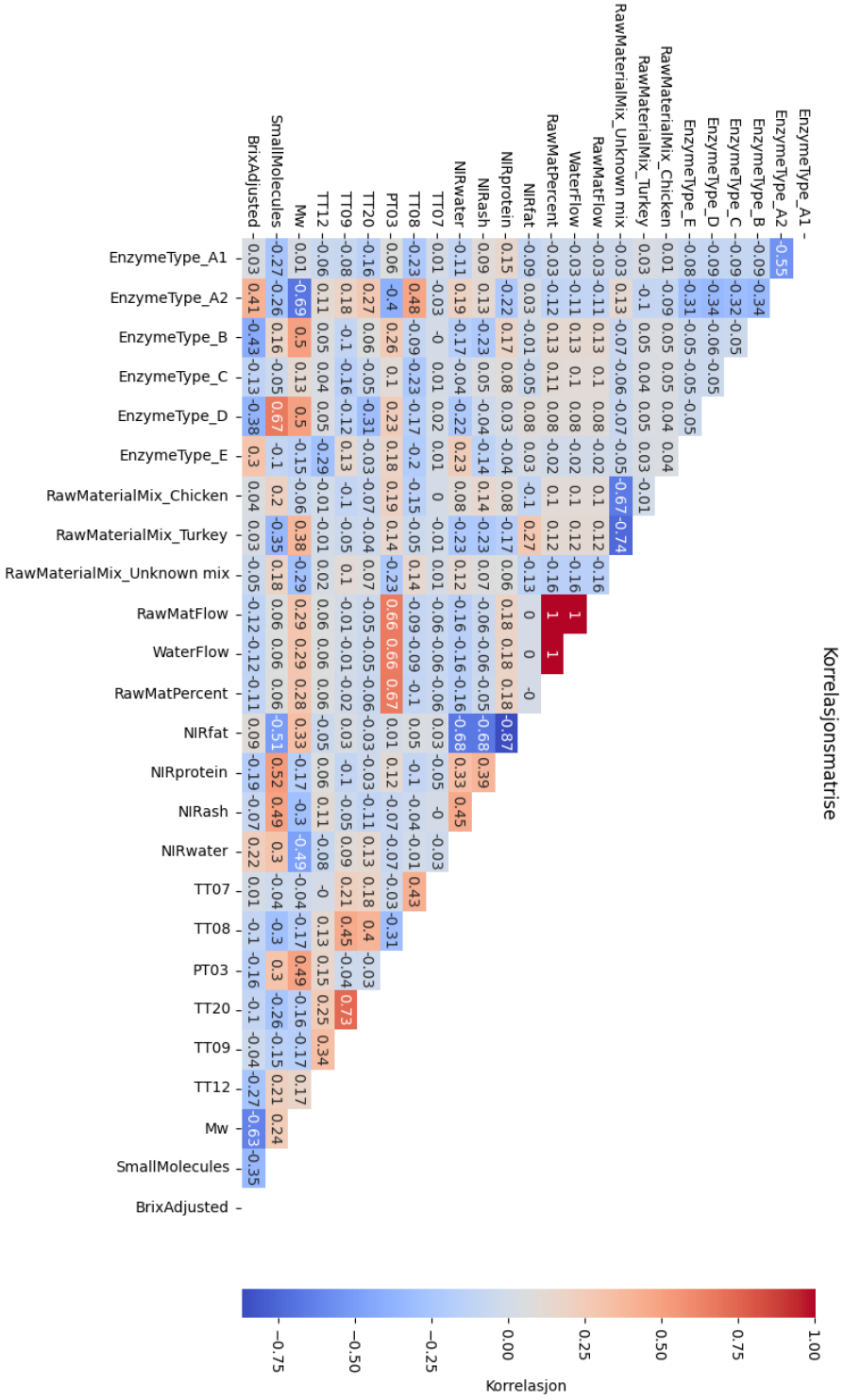
Tabell A.1: Oversikt over benyttede bibliotek og pakker til dataundersøkelse, dataprosessering, databehandling, valg av variabler, modellering og evaluering. I tillegg til forklaring av gjeldende versjoner og hensikt for bruk.

Navn:	**Versjon:**	**Hensikt:**
Hoggorm	0.13.3	PCA og PCR.
Hoggorm-plot	0.13.2	Visualisere resultater fra Hoggorm-pakken.
Matplotlib	3.7.1	Visualisering
NumPy	1.25.2	Håndtering av numerisk data, spesielt matriser og arrays, samt utføre matematiske beregninger.
Pandas	1.5.3	Importering av data av ulike datakilder. Manipulering og organisering av data med strukturer som "Series" og "DataFrame".
Scikit-learn	1.2.2	Python bibliotek for å importere maskinlæringsalgoritmer.
Seaborn	0.13.1	Visualisering
Missingno	0.5.1	Visualisering av manglende data.
Statsmodels	0.14.2	Visualisering av ACF plot.
Optuna	3.6.1	Hyperparameteroptimering
PyOD	1.1.4	Inspeksjon og behandling av ekstremverdier i datasettet.
LAMDA-SSL	1.0.2	Semi-veiledet regresjonsmodell CoReg.

A.4 Behandling av datasett

A.4.1 Korrelasjon

Korrelasjonsmatrise



Figur A.1: Korrelasjonsmatrise av rådata for utvelgelse av variabler.

Korrelasjonsbeskrivelse

Fra undersøkelsen, ble det observert flere sterkt korrelerte variabler, som vist i Figur A.1. Tabell A.2 viser en oversikt over variabler som er høyt korrelerte og satt til vurdering for ekskludering.

Tabell A.2: Oversikt over forklaringsvariabler og tilhørende sterkt korrelerte variabler.

Forklaringsvariabel:	**Korrelerte variabler:**	**Korrelasjonskoeffisient:**
RawMatFlow	RawMatPercent	1
WaterFlow	RawMatPercent	1
NIRprotein	NIRfat	-0.87
TT07	TT08	0.43
PT03	RawMatPercent	0.67
TT09	TT20	0.73

Etter ekskludering av høyt korrelerte og mindre viktige variabler, ble antall forklaringsvariabler redusert til 8.

Utvelgelse av korrelerte variabler

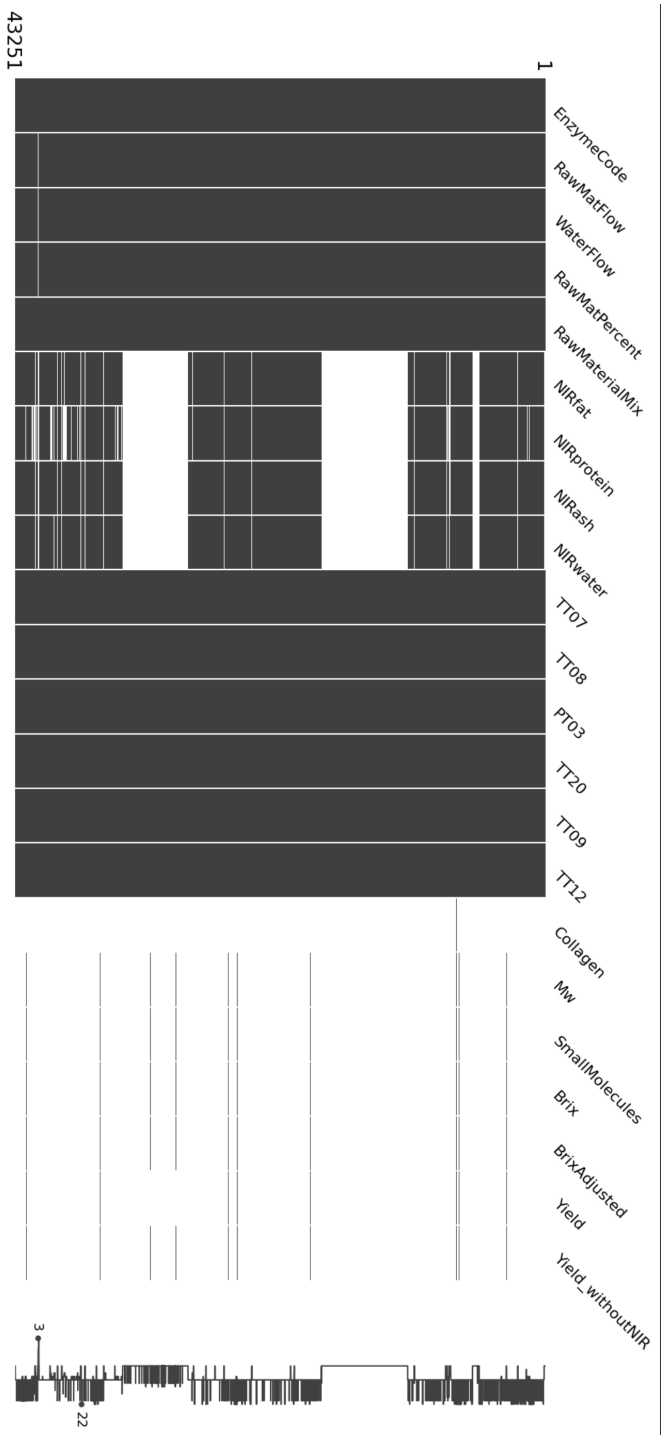
- "RawMatPercent" er kombinasjon av "RawMatFlow" og "WaterFlow". Variablene korrelerer sterkt og dermed ble det foretrukket å beholde "RawMatPercent" som består av informasjon fra de andre variablene.

- "PT03" er en variabel for trykkmåler. Den er ikke kontrollerbar, og er en konsekvens av strømningen som kontrolleres i begynnelsen av prosessen. "RawMatPercent" forklarer mye av informasjonen som dekkes av "PT03", dermed ble det besluttet å utelukke den sistnevnte forklaringsvariabelen.

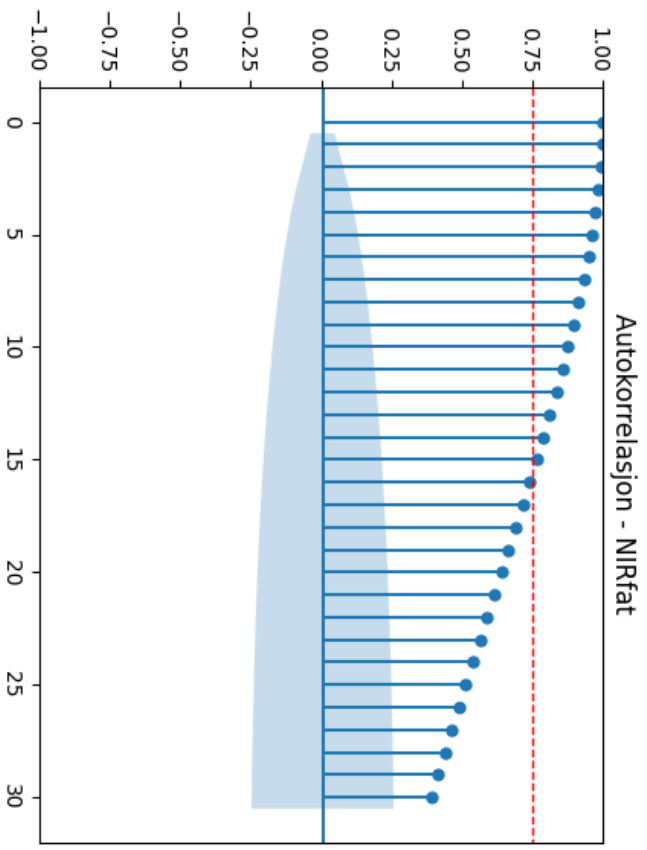
- Fra domenkunnskap om prosessen, var det kjent at "TT07" og "TT08" var etterfølgende temperaturmålinger, og hadde tilnærmet like målinger. "TT08" var foretrukket, da det målte temperatur ved innsettingen av enzym(er), motsetning til "TT07" som målte før. Som sterkt korrelerte variabler, ble "TT08" dermed foretrukket.

- "NIRprotein" og "NIRfat" var et høyt korrelert par av variabler. Da "NIRfat" hadde større variasjon i verdier enn protein, ble det foretrukket ovenfor "NIRprotein". I tillegg hadde "NIRprotein" flere manglende verdier sammenlignet med "NIRfat".

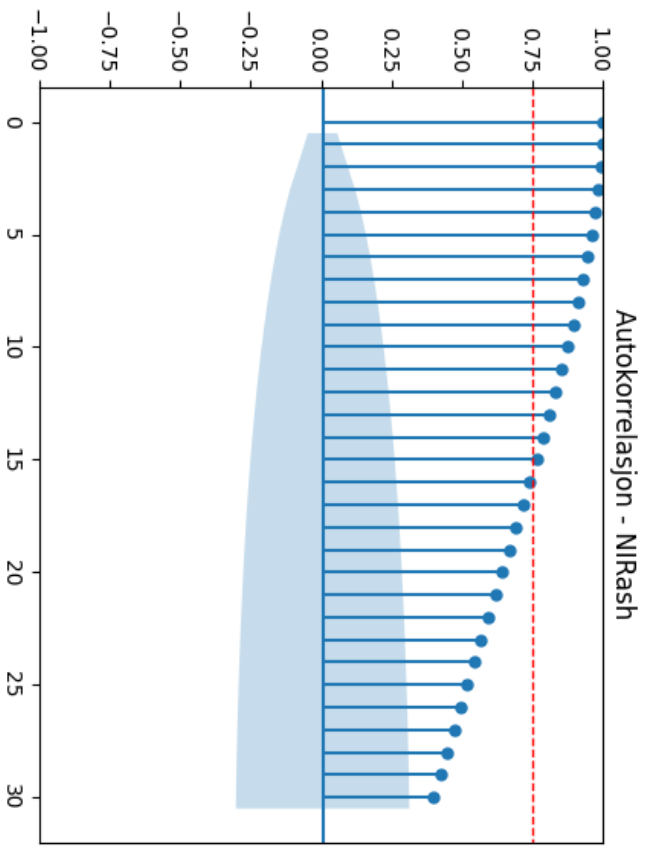
A.4.2 Behandling av manglende data



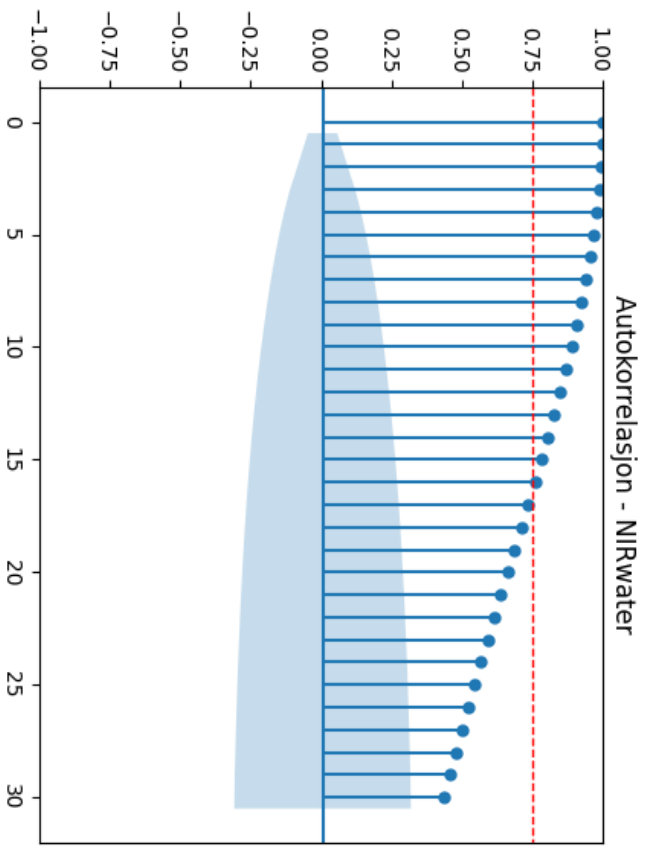
Figur A.2: Matrisen viser mønstre for de manglende verdiene for variablene i datasettet.



Figur A.3: Figuren viser autokorrelasjon for en kontinuerlige sekvens av observasjoner med fast tidsintervall for variabel NIRfat.



Figur A.4: Figuren viser autokorrelasjon for en kontinuerlige sekvens av observasjoner med fast tidsintervall for variabel NIRash.

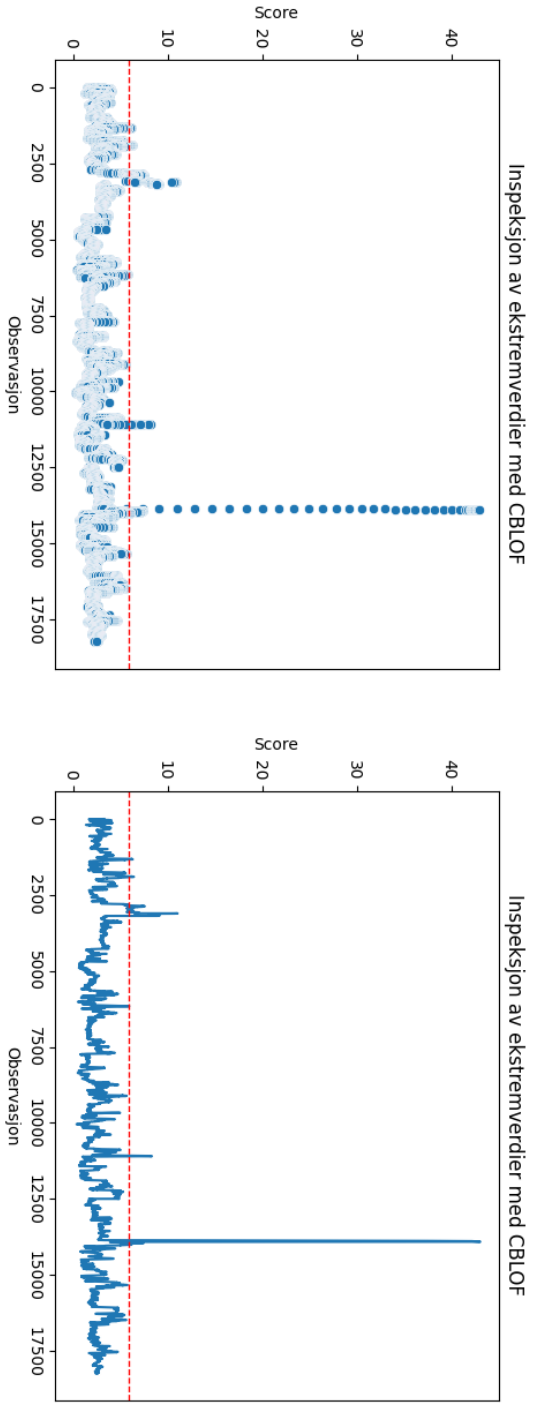


Figur A.5: Figuren viser autokorrelasjon for en kontinuerlige sekvens av observasjoner med fast tidsintervall for variabel NIRwater.

A.4.3 Behandling av kategorisk data

Etter encoding av kategoriske variabler, økte antall endelige forklaringsvariabler til 13.

A.4.4 Behandling av ekstrem data



Figur A.6: Visualisering av observasjoner og deres tilhørende score gitt av CBLOF. Høyere score indikerer sannsynlig ekstremverdi.

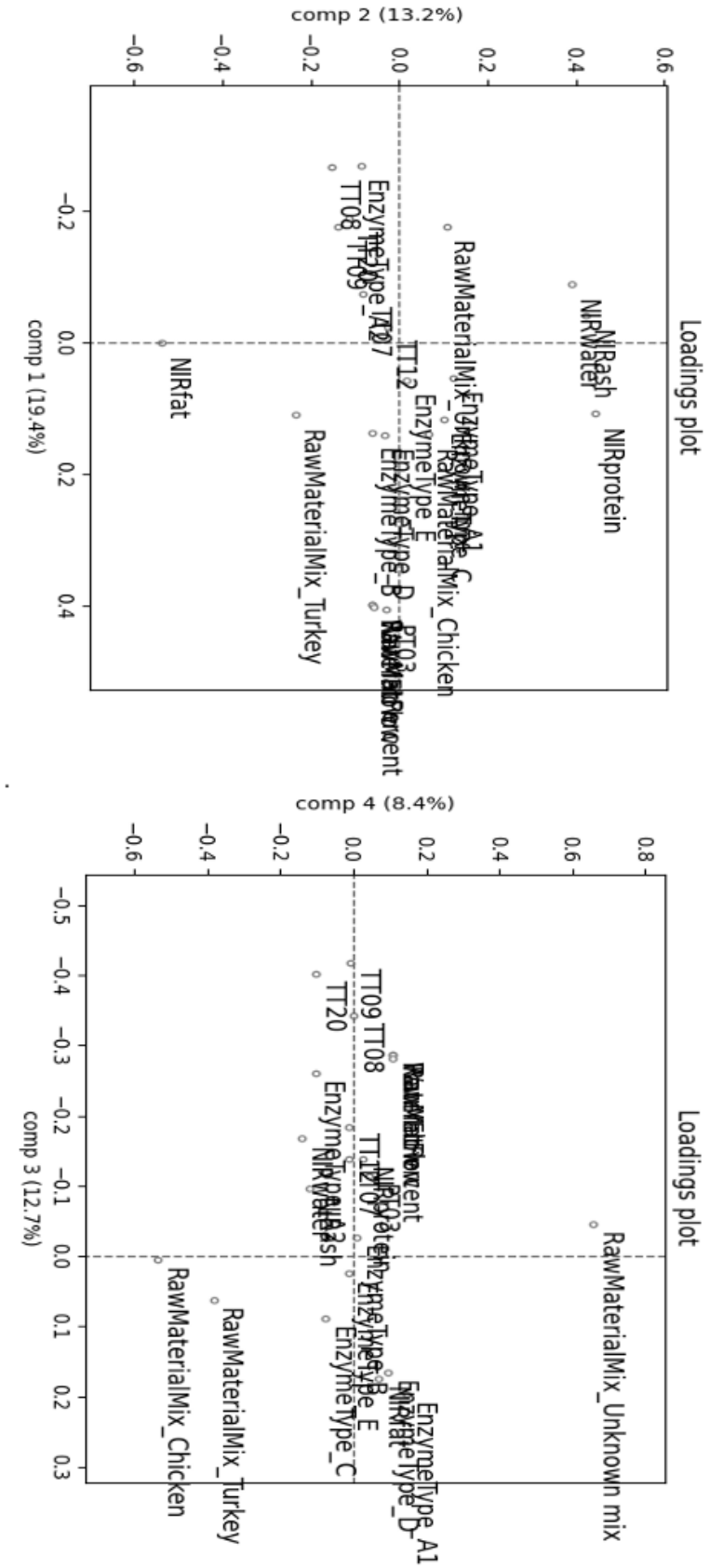
A.4.5 Inspeksjon og behandling av ekstreme verdier

CBLOF

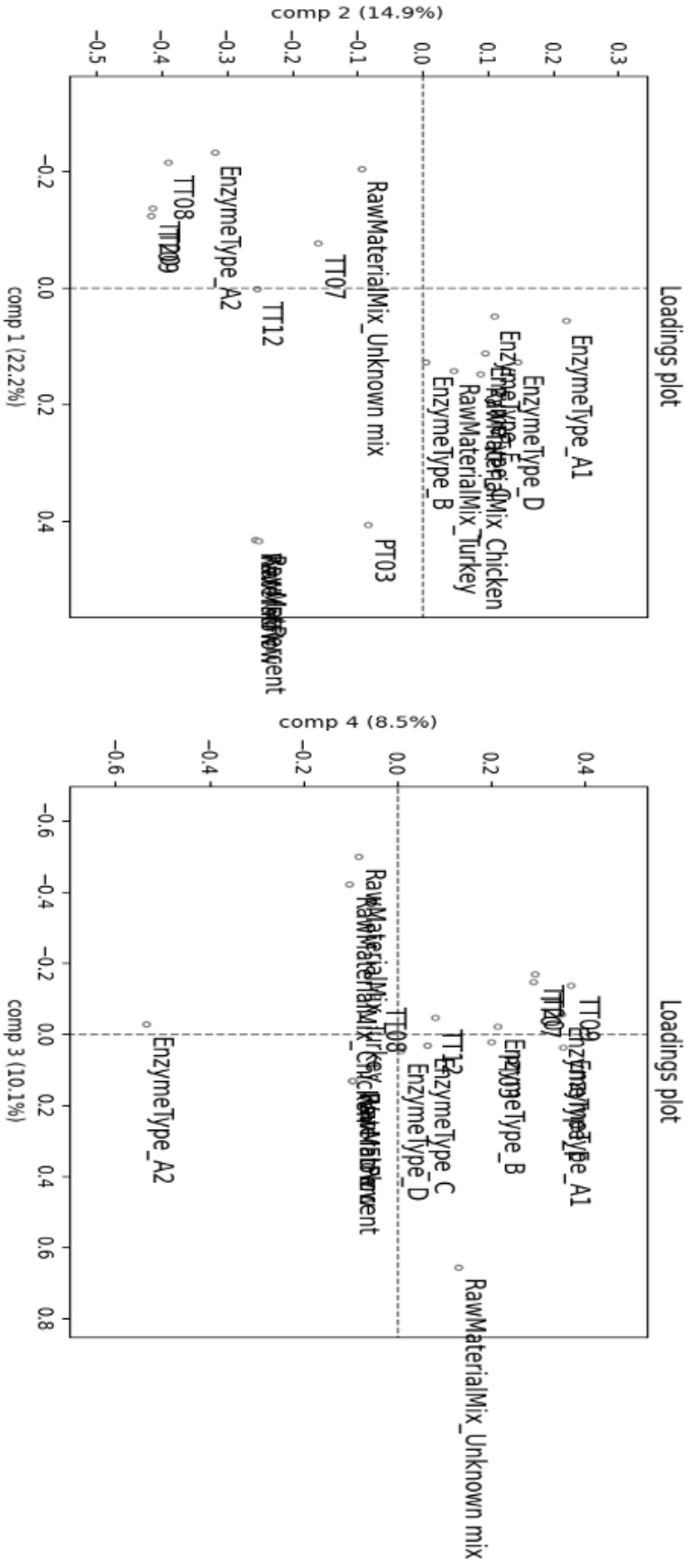
Etter behandling, har det endelige datasettet en dimensjon på (28 701, 16) med 445 markeringer for kvalitetsmålinger.

PCA-plot

PCA-analyse før behandling av data ble gjennomført ved å fjerne data med manglende verdier.



Figur A.7: PCA plott for komponent 1, 2, 3 og 4 før behandling.



Figur A.8: PCA plott for komponent 1, 2, 3, og 4 før behandling uten NIR.

A.4.6 Etterundersøkelse av data

Formålet med følgende seksjon er å presentere hvordan de ulike vurderingene og behandlingene har formet datasettet som skal benyttes til modellering.

Databehandlingen resulterte til datasett med dimensjon på (28 701, 17), derav 13 forklaringsvariabler og 3 responsvariabler.

A.5 Optimalisering med Optuna

A.5.1 Parametergrid for Optuna

Tabell A.3: Parametergrid for K-Neighbors Regressor (KNR).

Hyperparameter:	**Type:**	**Søkeområde:**
"n_neighbors"	Heltall	[1, 20]
"weights"	Kategorisk	["uniform", "distance"]
"p"	Heltall	[1, 10]

Tabell A.4: Parametergrid for Random Forest Regressor (RFR).

Hyperparameter:	**Type:**	**Søkeområde:**
"n_estimators"	Heltall	[100, 500]
"criterion"	Kategorisk	["squared_error", "absolute_error", "friedman_mse", "poisson"]
"max_depth"	Heltall	[1, 30]
"min_samples_leaf"	Heltall	[1, 5]
"max_features"	Flyttall	[0.1, 1.0, (step=0.001)]

Tabell A.5: Parametergrid for Støttevektor regresjonsmodell (SVR).

Hyperparameter:	**Type:**	**Søkeområde:**
"kernel"	Kategorisk	["linear", "poly", "rbf"]
"degree"	Heltall	[1, 20]
"gamma"	Kategorisk	["scale", "auto"]
"C"	Flyttall	[0.01, 100, (step=0.001)]
"epsilon"	Flyttall	[0.01, 100, (step=0.001)]

Tabell A.6: Parametergrid for Selvtrent-Random Forest Regressor (RFR).

Hyperparameter:	**Type:**	**Søkeområde:**
"n_estimators"	Heltall	[100, 500]
"criterion"	Kategorisk	["squared_error", "absolute_error", "friedman_mse", "poisson"]
"max_depth"	Heltall	[1, 30]
"min_samples_leaf"	Heltall	[1, 5]
"max_features"	Flyttall	[0.1, 1.0, (step=0.001)]
"maks_iterasjoner"	Heltall	[1, 20]
"std_terskel"	Flyttall	[0.01, 0.05, (step=0.001)]

Tabell A.7: Parametergrid for Coreg.

Hyperparameter:	**Type:**	**Søkeområde:**
"k1"	Heltall	[1, 10]
"k2"	Heltall	[1, 10]
"p1"	Heltall	[1, 10]
"p2"	Heltall	[1, 10]

A.6 Resultater

Tabell A.8: Gjennomsnittlige verdier og standardavvik for kvalitetsmåling Mw fordelt etter de kategoriske variablene for enzymtyper og råmaterialeblandinger.

Enzyme-Code	**Chicken (± std)**	**Turkey (± std)**	**Unknown mix (± std)**
A2	3790.49 (± 17.24)	5779.15 (± 132.36)	4895.03 (± 714.49)
A1	6115.61 (± 248.48)	8615.01 (± 239.19)	6987.64 (± 681.92)
B	8451.37 (± 258.31)	12040.17 (± 307.90)	9918.51 (± 623.11)
C	6306.28 (± 315.59)	10211.72 (± 487.11)	7740.60 (± 1649.36)
D	9203.36 (± 303.31)	11519.82 (± 231.24)	10106.26 (± 882.77)
E	5638.58 (± 122.40)	8032.60 (± 141.17)	6169.84 (± 575.29)

Tabell A.9: Gjennomsnittlige verdier og standardavvik for kvalitetsmåling BrixAdjusted fordelt etter de kategoriske variablene for enzymtyper og råmaterialeblandinger.

Enzyme-Code	**Chicken (± std)**	**Turkey (± std)**	**Unknown mix (± std)**
A2	0.137193 (± 0.002908)	0.140860 (± 0.001267)	0.114901 (± 0.020247)
A1	0.113128 (± 0.005199)	0.112524 (± 0.005287)	0.105053 (± 0.008737)
B	0.079697 (± 0.005482)	0.088499 (± 0.004244)	0.085117 (± 0.005915)
C	0.109651 (± 0.001726)	0.097053 (± 0.002137)	0.099491 (± 0.008216)
D	0.091478 (± 0.001389)	0.091670 (± 0.001910)	0.084312 (± 0.011814)
E	0.125589 (± 0.001438)	0.118578 (± 0.001950)	0.119577 (± 0.004752)

Tabell A.10: Gjennomsnittlige verdier og standardavvik for kvalitetsmåling SmallMolecules fordelt etter de kategoriske variablene for enzymtyper og råmaterialeblandinger.

Enzyme-Code	**Chicken (± std)**	**Turkey (± std)**	**Unknown mix (± std)**
A2	11.224147 (± 0.452722)	8.592447 (± 0.205506)	10.250605 (± 0.833181)
A1	11.786083 (± 0.530471)	9.277207 (± 0.091795)	10.423598 (± 0.765169)
B	12.898245 (± 11.968128)	9.347330 (± 0.526240)	12.024435 (± 0.906387)
C	11.764897 (± 11.101924)	8.745844 (± 0.389868)	11.058613 (± 1.218444)
D	14.328340 (± 13.962739)	12.462014 (± 0.463543)	14.216494 (± 1.094697)
E	11.408280 (± 10.996946)	8.476436 (± 0.208804)	10.777326 (± 0.727064)

A.7 Pseudokoder

Følgende seksjon vil inneholde pseudokoder som beskriver noen av algoritmene som ble benyttet i oppgaven...

Algorithm 6 COREG-LAMDA

Require: Markert datasett L, umarkert datasett U, antall nærmeste naboer k_1, k_2, maksimale iterasjoner i treningsfasen T, distansemetrikker p_1, p_2, poolstørrelse P

Ensure: Regressor $h^*(x) \leftarrow \frac{1}{2}(h_1(x) + h_2(x))$

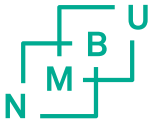
$L_1 \leftarrow L$; $L_2 \leftarrow L$
Lag en bufferpool U' fra U med størrelse P
$h_1 \leftarrow kNN(L_1, k_1, p_1)$; $h_2 \leftarrow kNN(L_2, k_2, p_2)$
repeat
    $stopp\_trening \leftarrow$ True
    $lagte\_indekser \leftarrow \emptyset$
    $fjernes \leftarrow \emptyset$
    for $j \in \{1, 2\}$ **do**
        $h \leftarrow h_j$
        $h' \leftarrow$ en kopi av h
        $L_X \leftarrow L_j$
        $L_y \leftarrow$ tilhørende etiketter
        for hver $x_u \in U'$ **do**
            $\hat{y}_u \leftarrow h.predict(x_u)$
            $\Omega \leftarrow h.kneighbors(x_u, k_j)$
            $L'_X \leftarrow L_X \cup \{x_u\}$
            $L'_y \leftarrow L_y \cup \{\hat{y}_u\}$
            $h'.fit(L'_X, L'_y)$
            $\Delta_{x_u} \leftarrow \sum_{x_i \in \Omega}((y_i - h.predict(x_i))^2 - (y_i - h'.predict(x_i))^2)$
        end for
        $sort\_indekser \leftarrow$ Sorter Δ i synkende rekkefølge
        $maks\_indeks \leftarrow$ første indeks i $sort\_indekser$
        if $\Delta_{maks\_indeks} > 0$ **then**
            $stopp\_trening \leftarrow$ False
            $lagte\_indekser \leftarrow lagte\_indekser \cup \{maks\_indeks\}$
            $x_u \leftarrow U'[maks\_indeks]$
            $\hat{y}_u \leftarrow h.predict(x_u)$
            $fjernes \leftarrow fjernes \cup \{maks\_indeks\}$
            if $j == 1$ **then**
                $L_1 \leftarrow L_1 \cup \{(x_u, \hat{y}_u)\}$
            else
                $L_2 \leftarrow L_2 \cup \{(x_u, \hat{y}_u)\}$
            end if
        end if
    end for
    if $stopp\_trening$ **then**
        break
    else
        $h_1.fit(L_1)$
        $h_2.fit(L_2)$
        $U' \leftarrow$ et nytt utvalg fra U med størrelse P
        $U \leftarrow U \setminus \{fjernes\}$
    end if
until T runder

Algorithm 7 Trening-validering-oppdeling med hensyn til dag og uke for kryssvalidering

```
function TRENING_VALIDERING_OPPDELING_DAG(data, n_splitt)
    treningssett_indeks ← dict()
    valideringssett_indeks ← dict()
    fold_rest ← n_splitt
    for i in range(n_splitt) do
        test_andel ← 1/fold_rest
        trening_markerte ← data
        val_markerte ← liste()
        data_markerte_filtrert ← data
        if i > 0 then
            for k in range(i) do
                data_markerte_filtrert ← data_markerte_filtrert.drop(valideringssett_index[k])
            end for
        end if
        for uke in data_unike_uker do
            for dag in data_unike_dager do
                data ← [(data_uke == uke) & (data_dag == dag)]
                test_rader ← math.floor((data.shape[0] * test_andel))
                test_data_dag ← data.head(test_rader)
                krav ← test_data_dag[RawMaterialMix"] =!"Unknown"
                if any in krav == True then
                    while krav.any() do
                        test_rader ← test_rader + 1
                        test_data_dag ← data.loc[krav.idxmax() :].head(test_rader)
                        test_data_dag ← test_data_dag.drop(index = krav.idxmax())
                        test_rader ← test_rader − 1
                        if (test_rader == 0) or (test_data_dag.shape[0] == 0) then
                            pass
                        else if (test_data_dag.shape[0] < test_rader) and (test_data_dag.index[−1] == data.index[−1]) then
                            test_rader ← test_rader − 1
                            test_data_dag ← data.head(test_rader)
                        end if
                        krav ← test_data_dag[RawMaterialMix"] =!"Unknown"
                        test_rader ← test_rader + 1
                    end while
                end if
                val_markerte ← pd.concat([val_markerte, test_data_dag])
                trening_markerte ← trening_data.drop(test_data_dag.index)
            end for
        end for
        treningssett_indeks[i] ← trening_markerte.index
        valideringssett_indeks[i] ← val_markerte.index
        fold_rest ← fold_rest − 1
    end for
    return treningssett_indeks, valideringssett_indeks
end function
```

Algorithm 8 Kalkulering av maks iterasjoner og pollstørrelse

```
1: function MAKS_POOL_ITER(X_trening_umarkert, tid_timer_maks, n_splitt,
   n_gjentagelser, n_runder)
2:     n_umarkert ← antall rader i X_trening_umarkert
3:     max_iter ← ⌊n_umarkert/2⌋
4:     pool_size ← ⌊max_iter/2⌋
5:     if pool_size > 250 then
6:         pool_size ← 250
7:     end if
8:     tid_grense ← (konverter tid_timer_maks til sekunder)
9:     if tid_timer_maks ≠ None then
10:        maks_iter_maks ← ⌊tid_grense/2⌋
11:    end if
12:    if n_splitt ≠ None then
13:        maks_iter_maks ← ⌊maks_iter_maks/n_splitt⌋
14:    end if
15:    if n_gjentagelser ≠ None then
16:        maks_iter_maks ← ⌊maks_iter_maks/n_gjentagelser⌋
17:    end if
18:    if n_runder ≠ None then
19:        maks_iter_maks ← ⌊maks_iter_maks/n_runder⌋
20:    end if
21:    if max_iter ≥ maks_iter_maks then
22:        max_iter ← maks_iter_maks
23:    end if
24:    return (max_iter, pool_size)
25: end function
```



Norges miljø- og biovitenskapelige universitet
Noregs miljø- og biovitskapelege universitet
Norwegian University of Life Sciences

Postboks 5003
NO-1432 Ås
Norway