

Användning av maskininlärning för att välja ut porträtt



LUNDS
UNIVERSITET

Lunds Tekniska Högskola

Department of Electrical and Information Technology
Faculty of Engineering, LTH, Lund University
SE-221 00 Lund, Sweden

Examensarbete:
David Axelsson Ahl
Lotta Paulsson

© Copyright David Axelsson Ahl, Lotta Paulsson

LTH Ingenjörshögskolan vid Campus Helsingborg
Lunds universitet
Box 882
251 08 Helsingborg

LTH School of Engineering
Lund University
Box 882
SE-251 08 Helsingborg
Sweden

Tryckt i Sverige
Media-Tryck
Biblioteksdirektionen
Lunds universitet
Lund 2018

ABSTRACT

This bachelor thesis evaluates the use of Clustering and Regression to separate good images from a given set of images. Whether the images are good or not is decided by the customers of the company that this system is developed for. In this thesis these customers are represented by a number of test persons. The images have been analysed with Amazon's service Amazon Rekognition where the answers are used as parameters in the JAVA-based machine learning tool WEKA.

The main goal of this thesis was to find parameters that are capable of selecting good images and also to gain knowledge in machine learning and algorithms. The thesis contains results from several tests where both Clustering and Regression are tested.

The result of this bachelor thesis is an evaluation that determine which of the methods Clustering and Regression, that are best suited to choose good images. The evaluation emerges into future areas of use for systems, like the system presented in this thesis. The result shows us that the learning capability of Regression is higher than it is for Clustering. Clustering on the other hand is more dependent of the ratio between number of parameters and quantity of data. The result also shows us that Clustering as well as Regression are able to choose good images with better results, than if they were randomly chosen.

the System

The system was developed in JAVA that only reached a development environment that managed to run the tests, meaning that no user interfaces was developed. The tests were executed from Eclipse.

Keywords

Clustering, Logistic Regression, Image Analysis, WEKA, Amazon Rekognition.

SAMMANFATTNING

Detta examensarbete utvärderar användandet av Clustering och Regression för att urskilja bra bilder från en mängd bilder. Vad som anses vara en bra bild avgörs av kunderna till företaget som systemet utvecklats för. I examensarbetet representeras dessa kunder av ett antal testpersoner. Bilderna har analyserats med Amazons tjänst Amazon Rekognition där svaren sedan används som parametrar i det JAVA-baserade maskininlärningsverktyget WEKA.

Syftet var att finna de parametrar och algoritmer som är mest intressanta för att välja ut bra bilder samt att utveckla kunskap och förståelse inom maskininläring. Examensarbetet innehåller resultat från en mängd test där både Clustering och Regression testas.

Resultatet av examensarbetet är en utvärdering av vilken utav algoritmerna Clustering och Regression som bäst lämpar sig för bildurval. Utvärderingen mynnar vidare ut i mer specifika användningsområden för ett system liknande det som presenteras i rapporten. Resultatet visar att Regression har bättre förmåga att lära sig än vad Clustering har. Clustering visar sig istället vara mer beroende av förhållandet mellan antal parametrar och mängde bilder. Vid användning av båda metoderna lyckas systemet välja ut bra bilder med högre sannolikhet än om samma bilder valts ut slumpmässigt.

Systemet

Systemet är utvecklat i JAVA och uppbyggt i en utvecklingsmiljö tillräcklig för att utföra testerna. Det utvecklades inte något användargränssnitt utan testerna genomfördes i Eclipse.

Nyckelord

Clustering, Logistisk Regression, bildanalys, WEKA, Amazon Rekognition.

FÖRORD

Vi vill tacka Elin Anna Topp på Lunds Tekniska Högskola och Magnus Lindgren, med medarbetare på NewSeed IT Solutions i Lund, för en lärorik och rolig tid tillsammans.

INNEHÅLLSFÖRTECKNING

Abstract	3
Sammanfattning	5
Förord	7
1 Inledning	11
1.1 Bakgrund	11
1.2 Syfte	12
1.3 Mål	13
1.4 Problemformulering	13
1.5 Motivering av examensarbetet	13
1.6 Avgränsningar	13
2 Teori	15
2.1 Maskininlärning	15
2.1.1 Övervakad Maskininlärning	15
2.1.2 Oövervakad Maskininlärning	20
2.2 Curse Of Dimensionality	23
3 Metod	25
3.1 Arbetsprocess	25
3.2 Instudering	26
3.3 Databas	26
3.4 Implementering	29
3.4.1 Flödesdiagram	29
3.4.2 UML-diagram	30
3.5 Integrering	31
3.5.1 Amazon Rekognition	32
3.5.2 Weka och ARFF	34
3.5.3 Clustering	37
3.5.4 Linjär Regression	38
3.5.5 Logistisk Regression	39
3.5.6 Antalet parametrar	39
3.6 Användartester	40
3.7 Verktyg	40
3.7.1 Euclidean Distance	40

3.7.2 Git	41
3.7.3 Weka	41
3.7.4 Amazon Rekognition	41
3.7.5 MySQL	42
3.8 Källkritik	43
3.8.1 WEKA	43
3.8.2 Amazon Rekognition	43
3.8.3 Maskininlärning	43
3.8.4 Val av träningsdata	43
4 Resultat	45
4.1 Användartester	45
4.1.1 Användartest efter 50 bilder	45
4.1.2 Användartest efter 500 bilder	50
4.2 Sammanställning av resultat	55
5 Diskussion och Analys	57
6 Slutsats	59
6.1 Vilka utav metoderna Regression och Clustering kan bäst få systemet att välja ut bra bilder?... 59	59
6.2 Vilka attribut gav bäst resultat?	59
6.3 Hur många parametrar behövde användas för att få bäst resultat?	59
6.4 Reflektion över etiska aspekter	60
6.4.1 SAMHÄLLSNYTTA	60
6.4.2 Etiska dilemman	60
6.5 Framtida utvecklingsmöjligheter	60
6.5.1 Förbättring	60
6.5.2 Användningsområden	60
7 Källförteckning	63
8 Terminologi	65

1 INLEDNING

I detta kapitel beskrivs bakgrund, syfte och mål för examensarbetet.

1.1 BAKGRUND

New Seed är ett konsultföretag som hjälper andra företag med IT-lösningar. Företaget har kontor i södra Lund, där examensarbetet utfördes. Vid vårt första möte med Magnus Lindgren på New Seed berättar han att de gillar utmaningar och gärna tar sig an svårare uppdrag. Han uttryckte intresse för att arbeta med maskininlärning och hade en idé hur det kunde appliceras på deras redan befintliga kund Elobina.

New Seed hjälper Elobina med att utveckla en webbtjänst för att göra fabrikstillverkning av tyg mer tillgänglig för deras kunder. Elobina erbjuder möjligheten att designa sina egna påslakan, kuddfodral eller köksdukar. Kunden kan skapa mönster men kan också välja att trycka upp en bild på textilen.

New Seed planerar att utveckla ett verktyg så att Elobinas kunder ska kunna skapa kollage för att trycka på textilier. Verktöget ska presentera ett antal bilder ur en bildsamling, till exempel från kundens egna instagram- eller facebook-konto, där personerna på bilderna uppfyller eller delvis uppfyller vissa kriterier. Exempel på kriterier är att personerna på bilderna ler eller har ögonen öppna. Hypotetiskt ska de kriterier som ställs på bilderna medföra att de bilder som presenteras, med högre sannolikhet, upplevs som bra bilder för kunden.

För att lära systemet som utvecklats under detta examensarbetet, som vidare kommer benämnas som systemet, vilka bilder som Elobinas kunder tycker är bra behövdes parametrar som kunde jämföras. Dessa parametrar hade New Seed tillgång till via Amazons tjänst Amazon Rekognition. Parametrarna visade på om det fanns en person på bilden, personens ansiktsuttryck och kvaliteten på bilden. Dessa parametrar användes i systemet som utvecklats under examensarbetet för att bestämma huruvida en bild var bra eller dålig. Generellt sätt uppfattas en bild som bra om personen på bilden ser glad ut. Denna definitionen för en bra bild tog examensarbetet avstamp i.

Vid inledningen av examensarbetet saknades kunskap om maskininlärning, därför inleddes studierna på en grundläggande nivå. För att få snabb överblick om vad maskininlärning är och vad det kan användas till spenderades tid till att se Youtube-videos och läsa introduktionskapitel i flertalet böcker om maskininlärning, artificiell intelligens och datasamling. Intresse låg i att utforska och jämföra

metoder inom övervakad- och oövervakad maskininlärning samt att utreda vilken metod inom dessa som var bäst lämpat för examensarbetet. Utifrån denna kunskap valdes Clustering och Regression som metoder att arbeta med. Clustering för att hitta dolda mönster i bilder som en människa inte direkt kan se. Regression valdes som övervakad inlärningsmetod för att kunna jämföra skillnaden i precision då systemet tränas av en människa jämfört med de dolda mönster som Clustering gav.

För att lära ett system vad som är en bra bild behövs träningsdata. Träningsdata lär systemet vilka bilder som ska klassas som bra och dåliga och därför är det viktigt att noggrant välja träningsdata. Om detta inte görs kommer systemet lära sig fel och bli oanvändbart. För att testa om ett system lärt sig rätt används testdata för att utvärdera hur bra systemet lärt sig. Testdatan är dock inte en del av träningsdatan.

Det JAVA-baserade maskininlärningsbiblioteket WEKA användes för implementation av maskininlärningen. Både algoritmer för Clustering och Regression kunde integreras i systemet med hjälp av WEKA.

Databasen utvecklades för systemet och implementerades av författarna till denna rapport. Efter att en bild analyserats med Amazon Rekognition erhöles värden som sparades i databasen. Dessa värden användes som parametrar till maskininlärningen. Även resultaten av maskininlärningen sparades i databasen.

Författarna till rapporten har själva implementerat och utformat systemet. Arbetet förväntas bidra till ökad kunskap inom maskininlärning för både New Seed och författarna. Förhoppningar finns även på att systemet kommer kunna användas av Elobinas kunder och för New Seeds framtida projekt.

1.2 SYFTE

Syftet med examensarbetet var att utreda vilken av metoderna Clustering och Regression som bäst kunde få systemet att välja ut bra bilder, givet det svar som fås efter analys med Amazon Rekognition. Examensarbetet ökade New Seeds övergripande kunskaper inom maskininlärning och har givit dem större möjligheter att använda sig av maskininlärning i framtida projekt.

1.3 MÅL

Systemet skulle med högre sannolikhet kunna välja ut bra bilder än om slumpvisa bildval hade gjorts. Detta har testats genom att låta testpersoner avgöra om bilden som presenterades av systemet var bra eller dålig. Systemet som utvecklades skulle bidra till att Elobinas kunder kunde skapa kollage på ett snabbt och roligt sätt.

1.4 PROBLEMFÖRMULERING

Examensarbetet behandlade följande problem.

1. Hur kan bra bilder särskiljas från dåliga?
2. Vilka utav metoderna Regression och Clustering kan bäst få systemet att välja ut bra bilder?
3. Hur många parametrar behövs för att kunna välja ut bilder så att Elobinas kunder reagerar positivt på dessa?
4. Vilka attribut är intressanta för att systemet ska välja ut bra bilder?

1.5 MOTIVERING AV EXAMENSARBETET

Examensarbetet skulle möjliggöra för Elobinas kunder att automatiskt ta fram bilder ur en bildsamling, som exempelvis instagram, för att skapa kollage. Detta för att Elobina vill utveckla och modernisera sin e-handel. Systemet skulle effektivisera Elobinas e-handel och göra den roligare genom att presentera de bästa bilderna från bildsamlingen som kunden sedan skulle kunna välja utifrån.

Examensarbetet har tagits fram utifrån intresset av att utforska maskininlärning men bidrar även till att New Seed erhåller kunskap som de senare kan applicera på framtida projekt.

1.6 AVGRÄNSNINGAR

Systemet som utvecklats under detta examensarbetet har inte utvecklats till en färdig produkt, utan endast för att utvärdera frågeställningarna under 1.4. De bilder som använts under examensarbetet begränsades till porträtt endast innehållande en person. Inlärningen tar bara hänsyn till ansiktets uttryck, inte till föremål eller hur fototekniskt bra en bild är.

I Systemet ingår inte någon egen implementation av bildanalys utan den görs av Amazons tjänst Amazon Rekognition. Maskininlärningsalgoritmerna implementeras inte från grunden istället används WEKAs bibliotek integreras

2 TEORI

I detta kapitel beskrivs teorin kring de metoder och algoritmer som används vid utveckling av systemet. Läsaren förväntas ha samma tekniska bakgrund som författarna.

2.1 MASKININLÄRNING

Maskininläring är ett område inom datavetenskapen som innebär att datorer, med hjälp av analys, lär sig att hitta mönster och strukturer i en datamängd. Inom maskininläring delas data ofta in i indata och utdata. Indata är den data som ska analyseras och utdata är resultatet av analysen.

Lärandet kan ske övervakat i form av träning, då används datapunkter där man redan känner till värdet på utdatan. Lärande kan också ske oövervakat, där algoritmer hittar dolda mönster och strukturer i en datamängd. När lärandet sker oövervakat behövs ingen träning. Examensarbetet har utforskat metoder inom övervakad- och oövervakad maskininläring.

2.1.1 ÖVERVAKAD MASKININLÄRNING

Inom övervakad maskininläring bygger lärandet på ständig inmatning av träningsdata och feedback till systemet. Feedbacken talar om för algoritmen för inläring hur bra det lyckats med sin analys. Träning av systemet görs genom att skicka in kategoriserad data till systemet och det är med hjälp av dessa två metoder som systemet lär sig.

Övervakad maskininläring bygger på ett invärde x och ett utvärde y . Dessa värden kan mappas med en algoritm. Målet är att på ett så precist sätt som möjligt lyckas approximera utvärde y för en ny instans. Algoritmen tränas med träningsdata tills att approximeringen har nått en tillräckligt bra nivå. Nivån för vad som anses tillräckligt bra varierar från system till system. Målet för detta system var att lyckas bättre än slumpvisa urval.

Vanliga metoder inom övervakad maskininläring är Regression och klassifikation. Regression kan även delas in i linjär- och logistisk Regression [5].

Klassifikation

Vid klassifikation sorteras instanser in i olika klasser. Utdatan tillhör en kategori, istället för ett numeriskt värde. Om ett fordon behöver klassificeras till att antingen vara en motorcykel eller en bil,

kan klassificering användas. Det gör det möjligt att förutse vilken typ kategori ett fordon ska tillhöra [6].

Regression

Vid Regression är utdatan ett realvärde, exempelvis ett pris eller en längd. Samtliga attribut är numeriska.

Linjär Regression

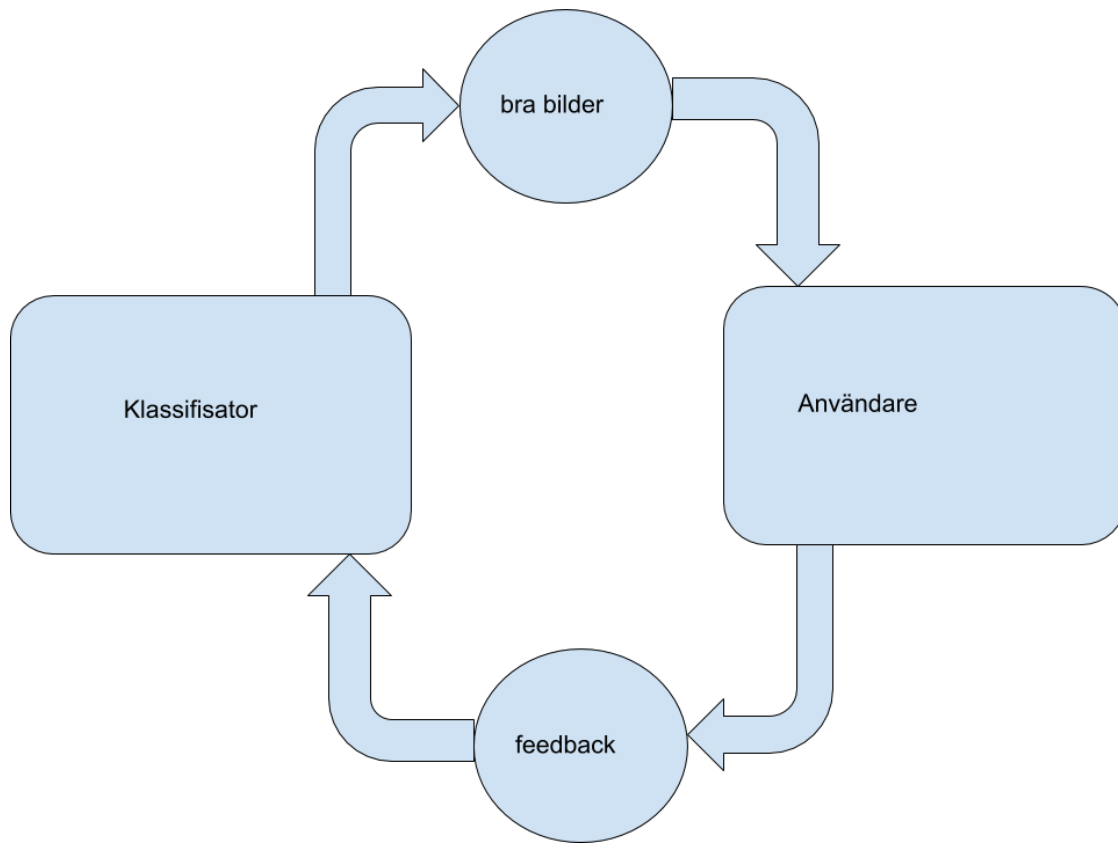
Linjär Regression passar bäst när samtliga attribut är numeriska. Grundtanken med linjär Regression är att få fram en linjär funktion som representerar medelvärden från indata. Den linjära funktionen används för att kunna förutse värden som ännu inte finns som indata. När funktionen byggs upp utifrån första instansen görs det på följande sätt [6]:

$$y = w_0 + w_1 a_1^{(1)} + w_2 a_2^{(1)} + w_k a_k^{(1)} \quad [6]$$

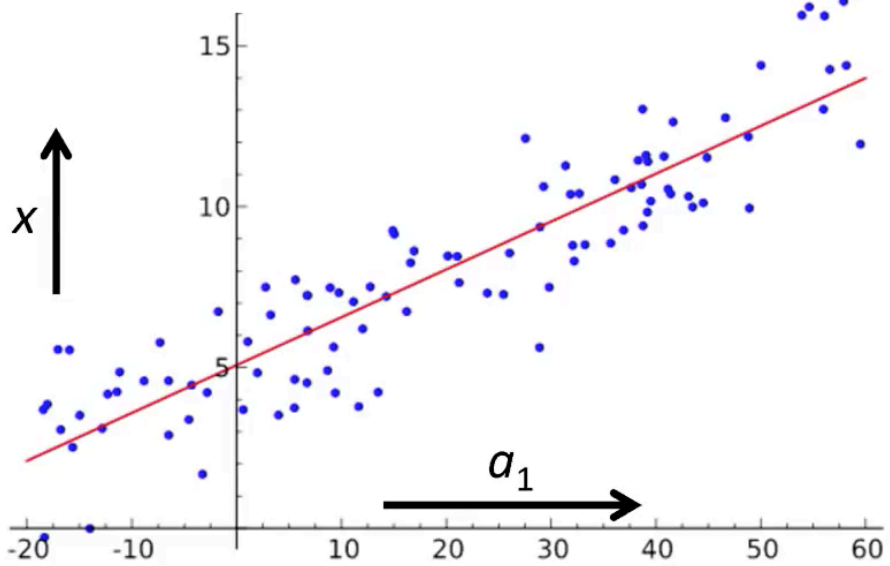
Funktionen (y) baseras på attribut (a) och vikt (w), där vikten avser hur stor påverkan varje attribut har. Ett attribut inom bildanalys är exempelvis om personen på bilden ler, och leendets värde.

Vikterna(w) beräknas utifrån träningsdata och kommer från feedback till systemet. Feedback kommer från användaren och är respons på resultatet av den analys som den linjära Regressionen har gjort, se *figur 1. Figur 2* visar en graf där den röda linjen representerar den linjära funktionen och de blå punkterna representerar mätdata. y -axeln som i detta fall representeras av värden på x , där värdet på x beräknas på följande sätt:

$$x = aw \quad [6]$$



Figur 1 visar träning med feedback.

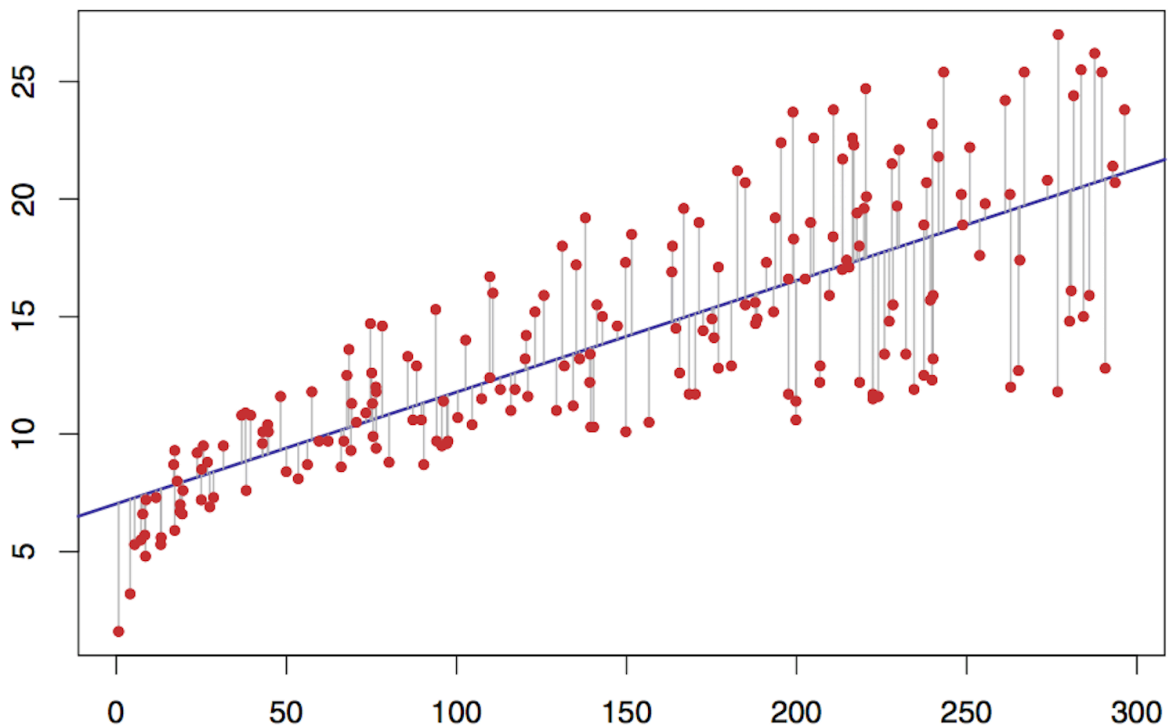


Figur 2 visar mätvärden och dess approximation i en graf.

För varje ny instans som läggs till görs en ny beräkning av den linjära funktionen. Approximering av den linjära funktionen görs genom att beräkna summan av felmarginaler för samtliga punkter i grafen.

Felmarginalerna består av skillnaden mellan den faktiska punktens värde och det förväntade värdet, se *figur 3*. För total approximation ska summan vara så liten som möjligt. Nedanstående formel används för att beräkna approximationen av den linjära funktionen:

$$\sum_{i=1}^n ((x^i) - (\sum_{j=0}^k w_j a_j^{(i)}))^2 [6]$$



Figur 3 visar avstånden mellan punkt och förväntat värde [17].

För att få en rättvis linjär funktion krävs det att avståndet mellan det faktiska värdet och det förväntade värdet är så litet som möjligt samt att summan av alla felmarginaler är minimal. När en funktion ger tillräckligt bra resultat kan den på ett fördelaktigt sätt användas för att approximera instanser där utvärden inte är kända sedan tidigare [6].

Logistisk Regression

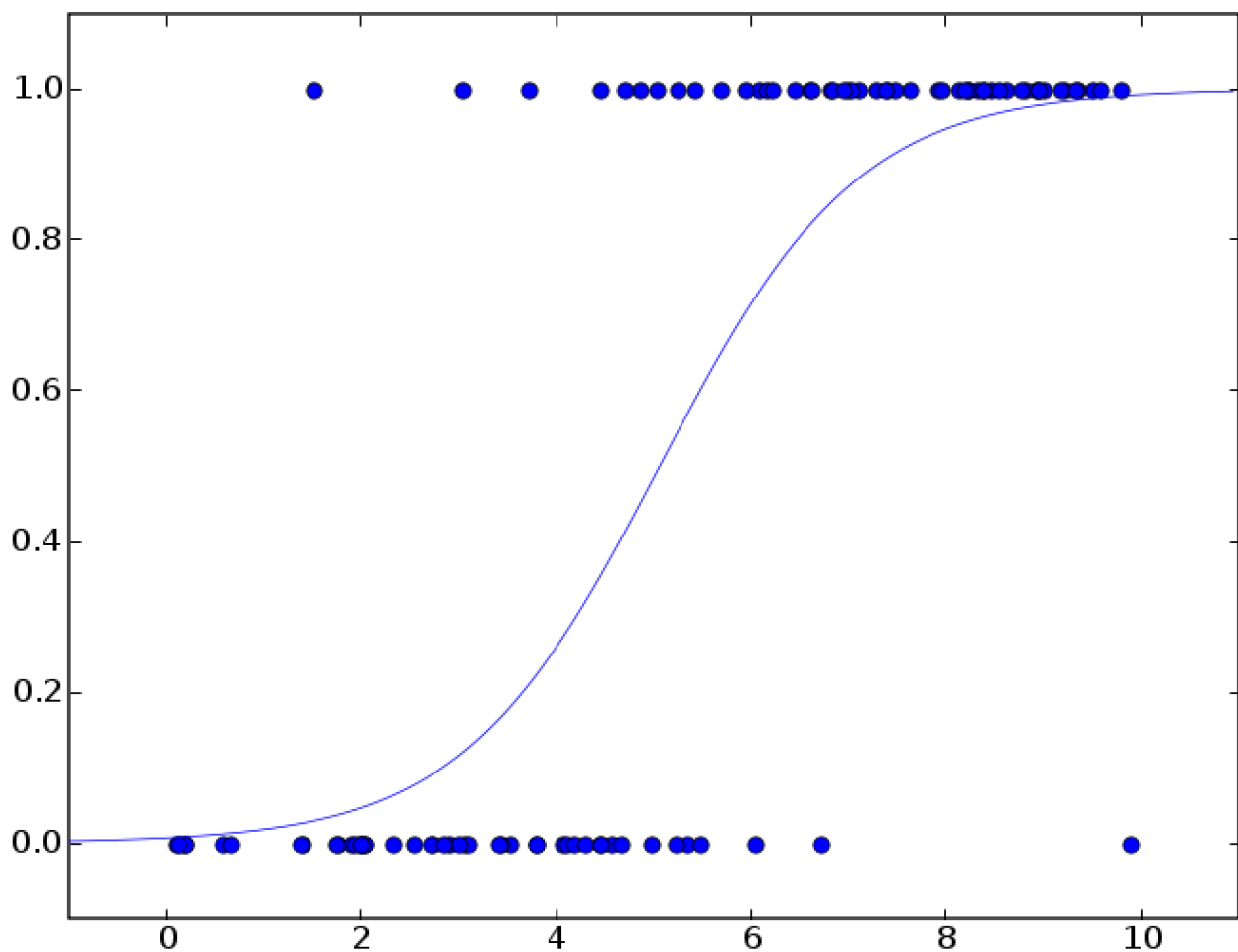
Logistisk Regression bygger på den linjära funktion som används vid linjär Regression. Istället för att approximera ett utvärde på det sättet som görs i linjär Regression så förutspås en klasstillhörighet.

Dock ger logistisk regression binär utdata vilket betyder att bara två utvärden kan erhållas. Detta gör att den logistiska regressionen är bättre lämpad för systemet.

Den linjära funktionen transformeras för att få fram den logistiska funktionen genom att använda nedanstående formel:

$$Pr[1|a_1 a_2 \dots a_k] = \frac{1}{(1 + \exp(-w_0 - w_1 a_1 - \dots - w_k a_k))} \quad [5]$$

Formeln transformerar alla värden till logistisk skala, där all utdata kategoriseras som ett eller noll. Formeln ger sannolikheten över vilken kategori värdet kommer att hamna i. I *figur 4* syns ett exempel på detta. Den logistiska Regressionen är användbar när utdata behöver kategoriseras till två kategorier och är även lämplig att använda när det behöver avgöras om en bild är bra eller dålig [5].



Figur 4 visar hur det logistiska kurvan ser ut och hur mätpunkter har kategoriserats som ett eller noll [18].

2.1.2 OÖVERVAKAD MASKININLÄRNING

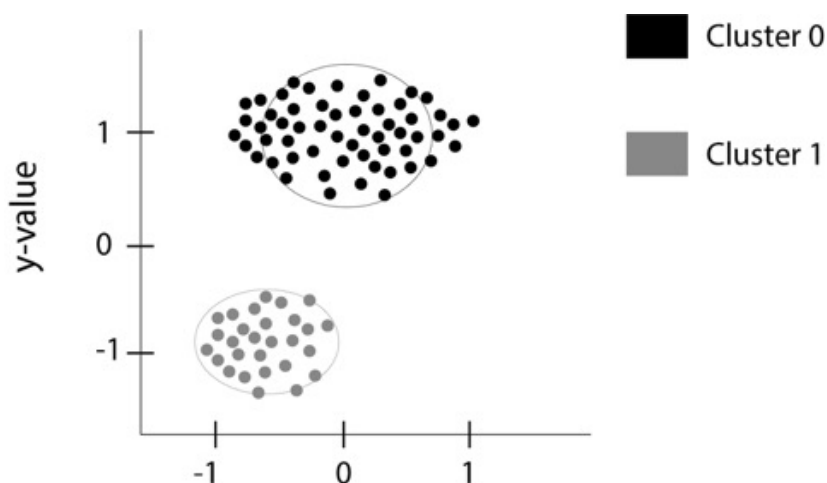
Oövervakad maskininläring är en gren inom maskininläring som varken kräver klassificering eller kategorisering utan istället hittar gömda mönster i en given datamängd.

Till skillnad från övervakad maskininläring så finns det ingen övervakare som förser oss med de korrekta svaren. Målet för en oövervakad inlärningsprocess är att hitta samband och mönster i den inkommande informationen. Man vill också få en uppfattning om vad som generellt brukar hända.

En metod för att hitta dessa samband och mönster är Clustering. Clustering kan till exempel användas för att gruppera ett företags kunddata i en flerdimensionell rymd där grupperna på ett eller flera sätt skiljer sig från varandra och där kunderna i en specifik grupp på ett eller flera sätt liknar varandra [5].

Clustering

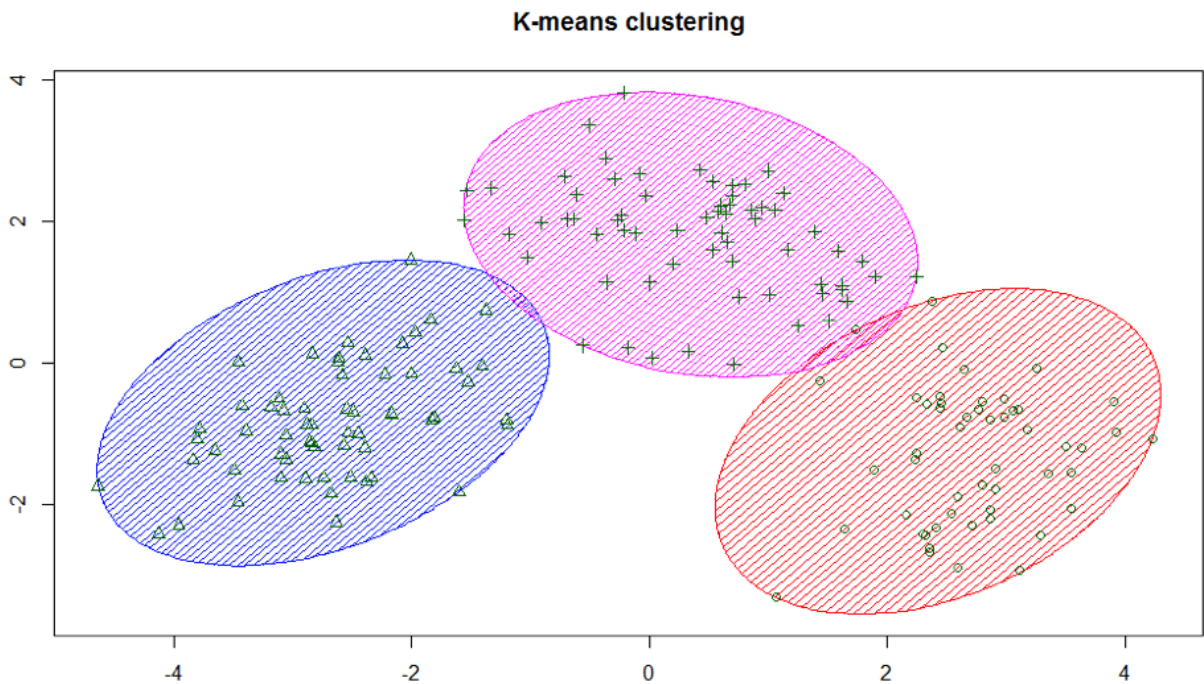
Clustering, som är en av de mest använda metoderna vid oövervakad maskininläring syftar till, som många andra metoder av det här slaget, att hitta ett strukturellt mönster i en ej klassificerad datamängd. Eftersom att det är subjektivt om en bild är bra skulle det möjligtvis finnas ett samband i bilderna som en människa ej kan se. Genom att använda Clustering skulle det hypotetiskt sett gå att gruppera bilder där en av grupperna tydligt består av bättre bilder [5][7]. Clustering med två parametrarna det vill säga två dimensioner kan visuellt representeras av en graf med två axlar se *figur 5*. Där idealfallet hade varit om de grå punkterna hade varit typiskt bra bilder och de svarta typiskt dåliga.



Figur 5 visar datapunkter indelade i kluster.

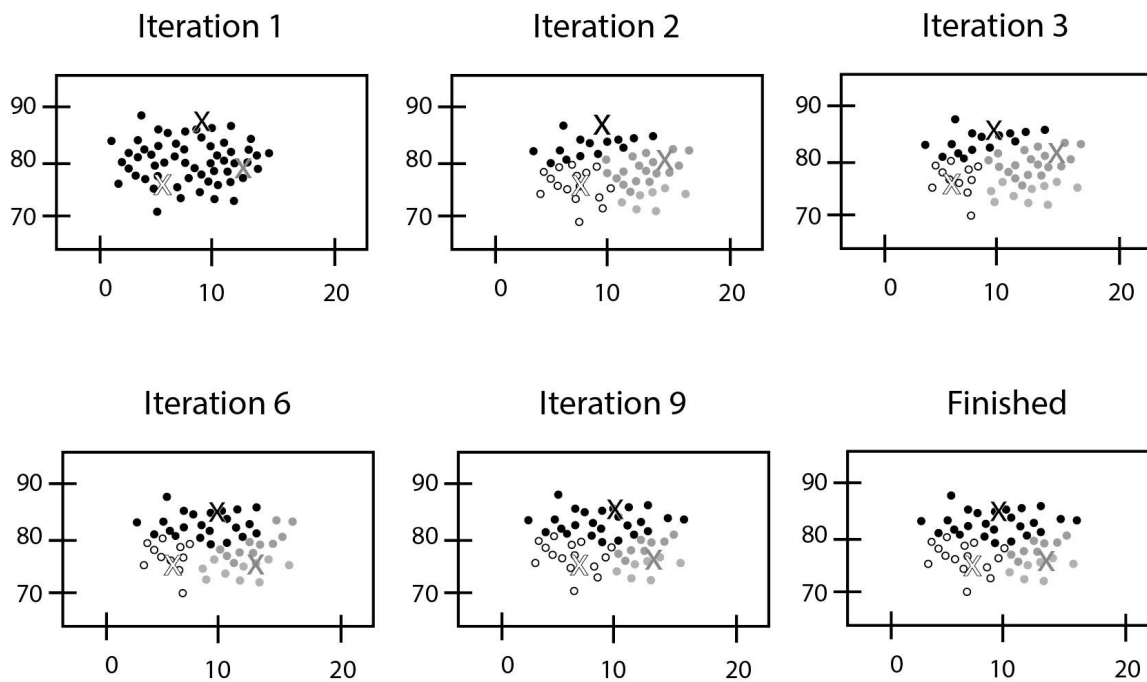
K-means

K-means är en av de Clustering-algoritmer som kan användas för att generera kluster. Detta går att göra i flera dimensioner och var därför ett alternativ eftersom att bra bilder kan bero på flertalet parametrar. K-means kan gruppera instanser i grupper och det görs baserat på avståndet till gruppens centroid som kan beskrivas som gruppens mittpunkt. Hur ett kluster där K-means använts visas i *figur 6*.



Figur 6 visar hur instansers mätvärden har fördelats i tre kluster [19].

För att bestämma ett klusters mittpunkt används en iterativ metod där man först väljer k antal godtyckliga instanser i ett dataset. Dessa instanser blir inledningsvis klustrets mittpunkt. “ k ” blir således antalet kluster i uträkningen. Fortsättningsvis tilldelas varje instans det kluster som ligger närmast. Avståndet till mittpunkten bestäms vanligtvis med Euklides formel. När alla instanser har tilldelats ett kluster beräknas varje klusters centroid ut vilket är slutet på en iteration. Denna process fortsätter tills dess att mittpunkterna ej förflyttas jämfört med föregående iteration se *figur 7* [7][11].



Figur 7 visar hur klusterna förändrar sig över ett antal iterationer iterationer. Det går att se hur instanserna har tillhört olika kluster fram tills att klusterna slutar förändras som händer mellan iteration 9 och Klar.

Vid användandet av K-means får alla instanser i datasetet en tillhörighet. När detta är fallet kommer instanser som ligger väldigt långt ifrån något av klusterna påverka ett klusters mittpunkt mer än önskvärt. I vissa fall skulle de vara mer önskvärt att utesluta instanser som ligger för långt ifrån något klusters mittpunkt [6].

Density Based Clustering

Till skillnad från K-means Clustering så baseras Density Based Clustering på densiteten i en modell istället för avståndet. Även då avståndsberäkning är en del av algoritmen så är det densiteten som är avgörande för vilket kluster som en instans kommer att tillhöra. Med densitet menas hur många punkter som ligger inom ett givet område. Om en instans befinner sig tillräckligt nära ett bestämt antal andra instanser räknas den in i det kluster.

För att avgöra om en instans kommer att ingå i ett kluster behövs två parametrar. Den ena anger maximala avståndet till en granne och den andra anger antalet grannar som måste finnas inom avståndet. Om båda villkoren är uppfyllda anges punkten som en kärnpunkt. Om en instans inte har tillräckligt många grannar anges instansen som en gränspunkt, vilket är en datapunkt som utgör gränsen för var ett kluster slutar. Om inget av villkoren uppfylls anges punkten som oljud och får ingen klustertillhörighet.

Fördelar med Density Based Clustering är att kluster som skapas inte alltid blir konvexa. Ett kluster kan istället anta en godtycklig form. Density Based Clustering kan också identifiera oljud och kan därför bli av med extrema instanser [7].

2.2 CURSE OF DIMENSIONALITY

För att kunna bibehålla inläringens prestationsnivå krävs i allmänhet låg dimensionalitet kombinerat med stora mängder data. För varje dimension som adderas ökar mängden data exponentiellt. Om det finns 10 datapunkter som används för en dimension, innebär det att det behövs 100 datapunkter för två dimensioner för att samma precision ska behållas. Om mängden data är för liten i förhållande till antal dimensioner fås otydliga resultat [8].

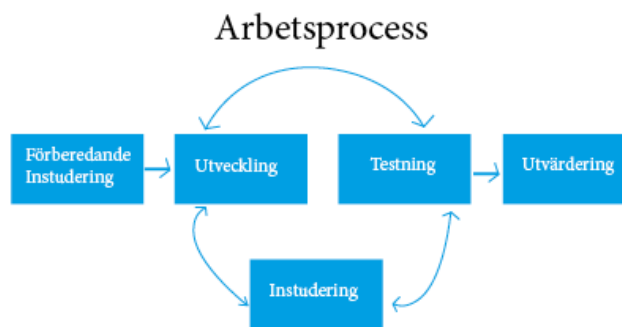
Curse Of Dimensionality hade påverkan på examensarbetet. Testerna för en dimension utfördes på 50 datapunkter. För att få rättvisande resultat behövde egentligen testerna som gjordes för två dimensioner innehålla 2500 datapunkter, eftersom mängden datapunkter ökar exponentiellt för varje dimension som adderas. Vid närmandet av testerna visade det sig inte möjligt att få tag på 2500 bilder. Testerna utfördes på 500 bilder, vilket var den mängden bilder som till slut erhöles innan testerna var tvungna att göras. Hänsyn till Curse Of Dimensionality tas vid analys av resultatet [15][8].

3 METOD

I detta kapitel beskrivs utvecklingen av systemet och vilka tredjepartsverktyg som integrerats. Kapitlet behandlar också utvecklandet av en databas som ligger som grund för systemet och arbetet.

3.1 ARBETSPROCESS

Arbetsprocessen kan delas in i fyra delar, instudering, design av databas, implementering och integrering samt testning. Dessa fyra delar har varit sammanvävda under arbetsprocessen eftersom det har varit en iterativ process. *Figur 8* visar var iterationerna har tagit plats.



Figur 8 visar examensarbetets arbetsprocess

Examensarbetet har utförts på NewSeed i Lund. NewSeed har erbjudit kontorsplats och en server där bilder och databas kunde lagras. Att vara på plats på NewSeed underlättade speciellt när användartester gjordes men också när datatekniska problem uppstod då det snabbt gick att få handledning.

Examensarbetet skulle skapa ett system som skulle lära sig kontinuerligt, det vill säga för varje uppladdad bild skulle precisionen öka. Med detta i åtanke valdes Clustering som en av metoderna som examensarbetet skulle använda sig av. Jämförelse av en övervakad med en oövervakad inlärningsmetod gjordes med förhoppning att få en tydlig skillnad i resultat mellan metoderna, men också utav intresset att förstå skillnader mellan de olika grenarna inom maskininlärning.

Examensarbetets bas ligger i den databasstruktur som utvecklats. Databasen byggdes upp utifrån de svar som Amazon Rekognition kunde ge eftersom att de var dessa svar som skulle lagras. De lagrade svaren behövdes också för att bygga upp ARFF-dokumentet som användes för att analysera med WEKA vilket beskrivs senare i avsnitt 3.5.2. För att köra WEKAs algoritmer i detta examensarbete inkluderades dem i det lokala javabiblioteket vilket möjliggjorde att dem kunde anropas från den JAVA-kod som implementerats. Den analysen som gjordes av WEKAs algoritmer sparades i databasen och i HTML-filer som kunde presenteras i en webbläsare.

För att kunna avgöra om examensarbetet lyckats behövdes något mätbart test, vilket redan tas upp i avsnitt 1.3. Testet utformades på ett sätt så användarna kunde verifiera om bilderna som togs fram var bra bilder, se avsnitt 4.1.

3.2 INSTUDERING

Clustering kan användas för att hitta dolda mönster i en given datamängd och för att klassificera den, därför ses Clustering som ett bra alternativ för detta examensarbete. Regression skulle däremot kunna reagera på vad flertalet användare gillar och lära sig utifrån det. Genom att utforska dessa två metoder skulle det gå att avgöra skillnader i precision för de olika metoderna.

3.3 DATABAS

Databasen är designad för att spara bildinformation. Designen har vuxit fram under arbetet och ändrats flertalet gånger. Den slutgiltiga designen är framtagen med motivering att det är bättre att hålla mer information än man initialt behöver för att möjliggöra framtida utvecklingsmöjligheter.

Databasen skapades delvis för att Amazon Rekognition är avgiftsbelagd och delvis för att det tar tid att skicka bilder på analys. Därför borde bara bilder som aldrig analyserats skickas till Amazon Rekognition för analys.

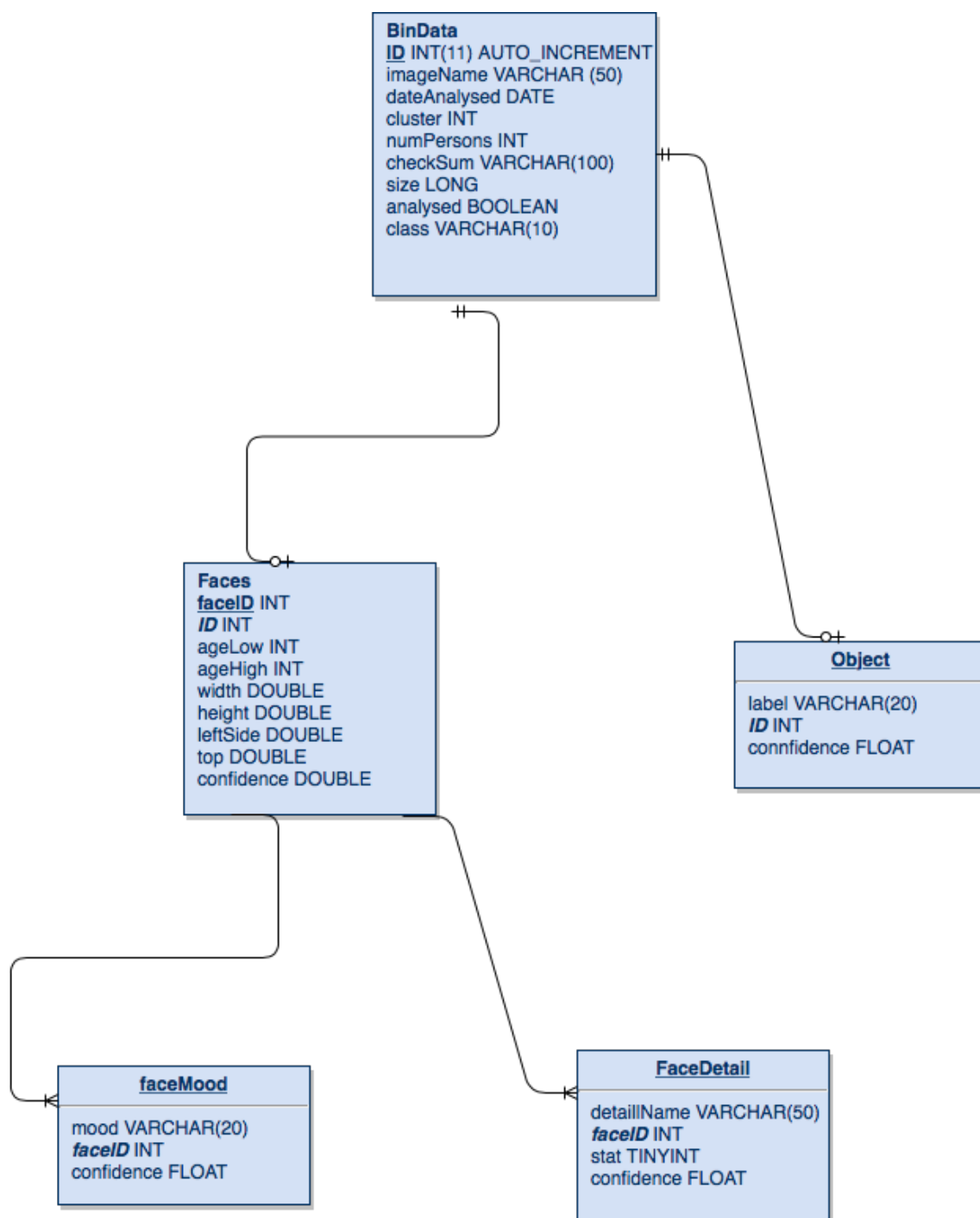
Bildinformationen som sparas i databasen är svaren som fås efter att bilderna har analyserats med Amazon Rekognition. Information som fås efter en analys är:

- De objekt som ingår i en bild.
- Ansiktsdetaljer och känslor som är kopplade till ansiktet.
- Om det finns kända personer på bilden.
- Om bilden innehåller någon typ av text.

För detta examensarbetet används information från FaceDetail och faceMood, resterande data sparades ifall en annan typ av analys vill göras eller ifall en vidareutveckling ska göras. Ett diagram över databas visas i *figur 9*.

Själva bilden sparas ej i databasen som binärdata, då det snabbt skulle göra databasen långsam i takt med att fler bilder läggs in i systemet. Därför utformades istället ett separat filsystem där bilderna länkas med hjälp av ID-numret som sparas i Tabellen BinData.

I tabellen BinData lagras bland annat ett unikt ID, bildens storlek och en checksumma. Checksumman beräknas med avseende på bildens storlek. När en bild ska analyseras görs en kontroll på om checksumman och bildstorleken redan finns för en bild i databasen. Detta kommer säkerställa att samma bild aldrig läggs in två gånger. Tyvärr finns risken att flera olika bilder får exakt samma storlek och checksumma, den är dock liten.

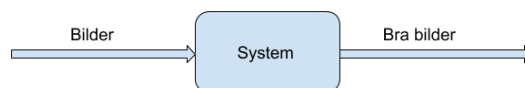


Figur 9 beskriver hur informationen lagras i databasen. Bindata representerar en bild, där lagras detaljer som exempelvis klass, kluster och antal personer som bilden innehåller. Varje bild kan innehålla ansikten och objekt dessa sparas i Faces samt Objects. Varje ansikte har ett antal ansiktsdetaljer såsom Smile, eyse Open och Beard dessa sparas i FaceDetail. Till varje ansikte identifieras också ett antal känslor, dessa sparas i FaceMood

I faces sparas ansiktets position som beskrivs av leftSide, top, width och height. leftSide indikerar var i bilden ansiktets vänstra sida befinner sig i bilden. Top indikerar var ansiktets översta del befinner sig i bilden. Sedan används width och height för att bestämma hur långt ansiktet sträcker sig över bilden. Denna information används inte i examensarbetet men skulle kunna användas om en utveckling av systemet görs.

3.4 IMPLEMENTERING

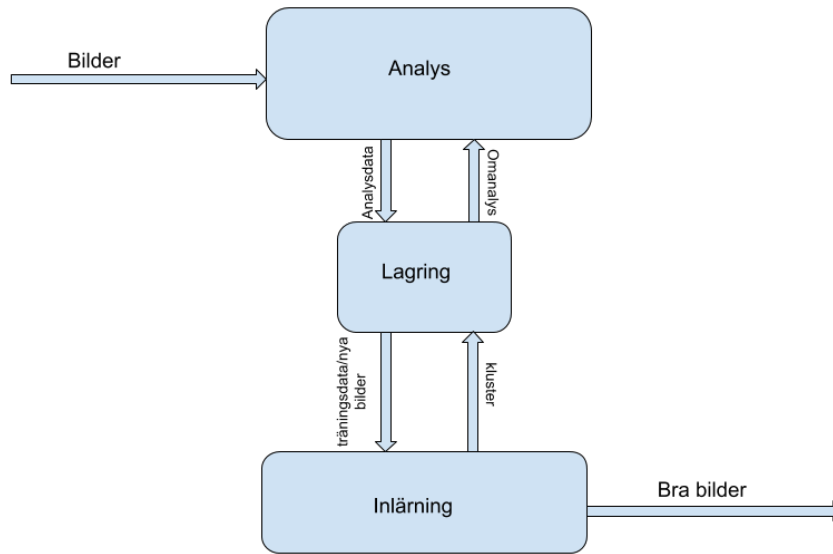
Utifrån ses systemet som en svart låda där en eller flera bilder kan matas in och undersöks, se *figur 10*. Resultatet av undersökningen visar vilka bilder i bildsamlingen som kommer uppfattas som bra.



Figur 10 visar systemets funktion

3.4.1 FLÖDESDIAGRAM

Alla bilder analyseras först av Amazon Rekognition för att sedan lagra värdena i databasen. Därefter hämtas värdena från databasen och används för inläring och träning. När bilderna körs i inläringen lagras klustertillhörigheten i databasen. De bilder som väljs ut som bra bilder hämtas i en mapp på datorn, via bildens ID i databasen och sparas i en HTML-fil. Dessa anrop görs ifrån de exekverbara programmen i testmiljön i Eclipse. Anropen visas i *figur 11*.



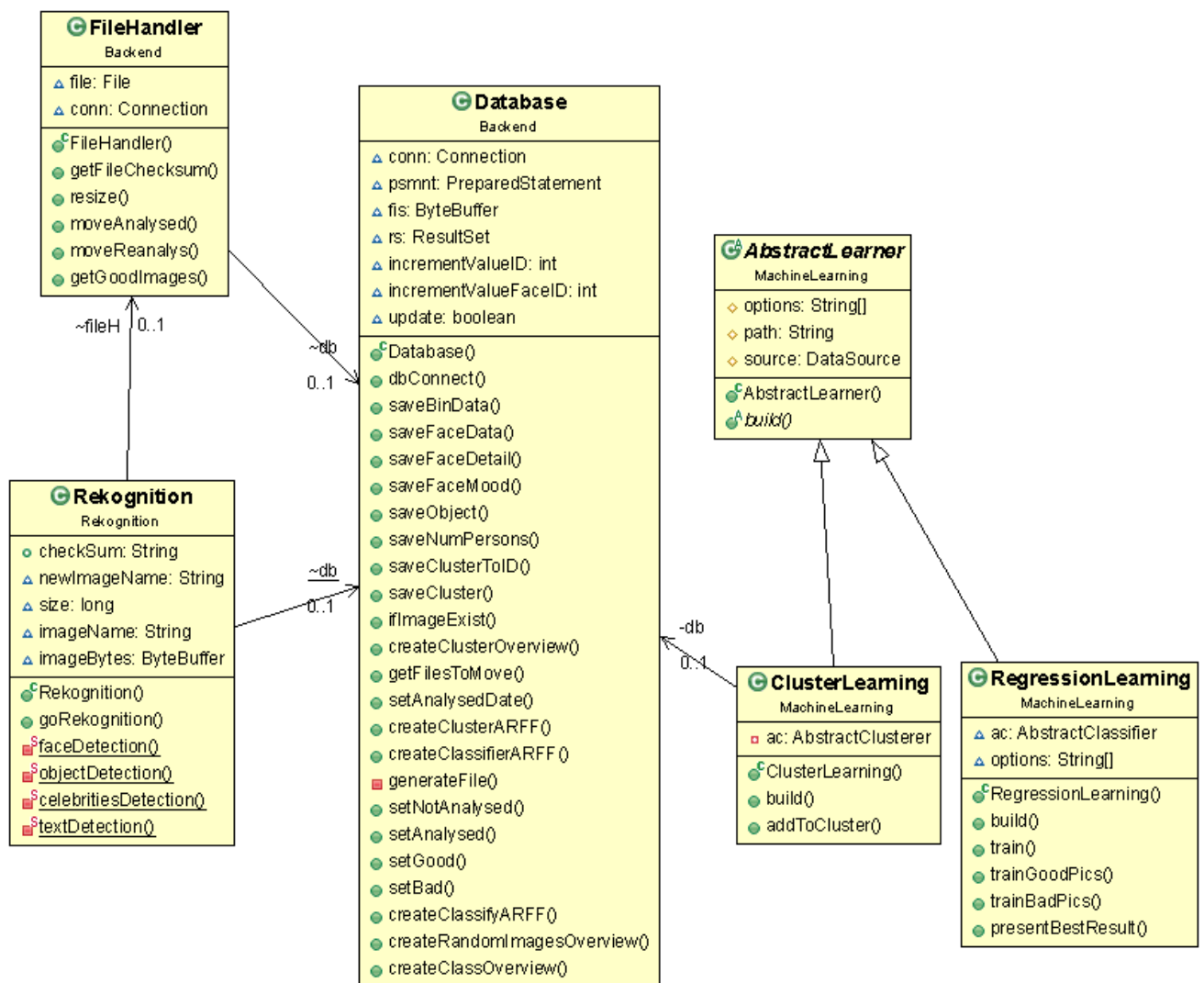
Figur 11 visar hur bra bilder väljs ut

3.4.2 UML-DIAGRAM

Systemet är fördelat i tre paket; Rekognition, Backend och MachineLearning, se figur 12. I paketet Rekognition finns endast en klass, Rekognition. Rekognition innehåller integreringen av Amazons tjänst "Rekognition" för bildanalys. I ovan nämnda klass görs även anrop till databasklassen, vilken lägger in värdena från bildanalysen i databasen. För att kunna köra Amazon Rekognition behövs användarnamn och lösenord till Amazon Web Service samt import av två stycken jar-filer.

Paketet Backend består av två klasser; Database och FileHandler. Databasklassen består av ett antal metoder som skriver till eller från databasen. FileHandler-klassen har ett flertal viktiga funktioner, bland annat flyttar den analyserade bilder mellan mappar på datorn och minskar även en bilds storlek innan den läggs in i databasen.

Paketet MachineLearning innehåller tre klasser; AbstractLearner, ClusterLearning och RegressionLearning. Dessa klasser sköter själva inläringen. De läser in en ARFF-fil (se avsnitt 3.5.2) som sedan blir en instans, som sedan blir data till maskininlärningsalgoritmerna.

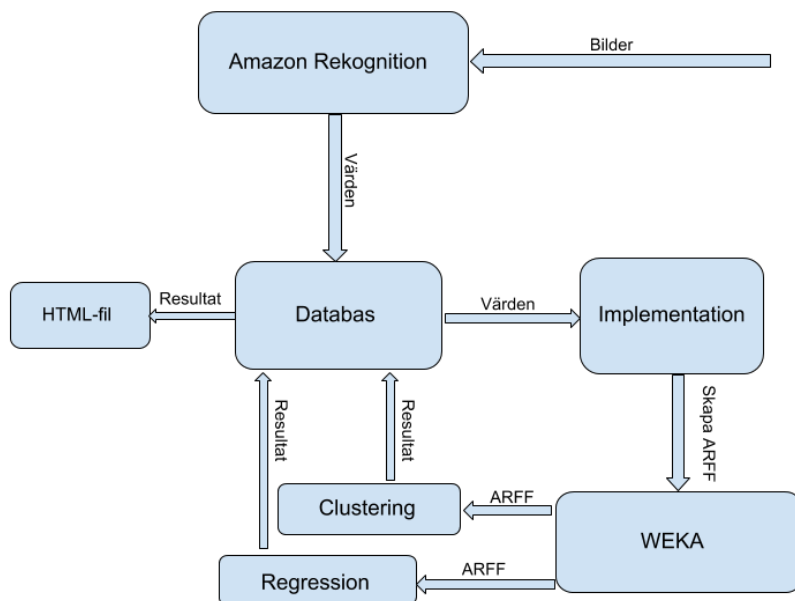


Figur 12 visar klassdiagram

3.5 INTEGRERING

I det här avsnittet beskrivs hur algoritmerna för Clustering, logistisk Regression och tjänsten Amazon Rekognition samt tredjepartsverktygen WEKA har integrerats i systemet och hur problem har lösts.

Systemet är uppbyggt utifrån egen implementation av databasen. Maskininlärningen som sker med hjälp av WEKAs bibliotek och bildanalysen som görs av Amazon Rekognition har båda integrerats i systemet. I implementationen finns metoder för att skapa ARFF-filer av värdena i databasen. *Figur 13* beskriver hur Amazon Rekognition, databasen, implementationen och WEKA integrerar med varandra.



Figur 13 visar systemet och hur Amazon Rekognition, databas, den egna implementation och WEKA interagerar med varandra

3.5.1 AMAZON REKOGNITION

Till Amazon Rekognition skickas de bilder som ännu inte analyserats och tillbaka kommer information om ansikten, känslor, objekt och text. Bilderna skickas till amazon i en bitström och tillbaka kommer informationen som kan ses i *figur 14*. Den information som sparas i systemet är ansiktsdetaljer som leende, öppna ögon och känslor. De objekt som finns med på bilderna sparas också. I avsnitt 3.4 beskrivs hur databasen byggts upp och hur allting lagras.


```

The detected face is estimated to be between 19 and 36 years old.
Here's the complete set of attributes:

"boundingBox"
"width" : 0.58625,
"height" : 0.7816667,
"left" : 0.086875,
"top" : -0.0125

"ageRange"
"low" : 19,
"high" : 36

"smile" :
"value" : true,
"confidence" : 85.668655

"eyeglasses" :
"value" : false,
"confidence" : 95.25664

"sunglasses" :
"value" : false,
"confidence" : 98.98994

"gender" :
"value" : "Male",
"confidence" : 99.631226

"beard" :
"value" : true,
"confidence" : 61.823742

"emotions" :
"type" : "HAPPY",
"confidence" : 93.65743

"type" : "CONFUSED",
"confidence" : 17.817127

"type" : "ANGRY",
"confidence" : 0.67237616

"landmarks" :
"type" : "eyeLeft",
"x" : 0.26456237,
"y" : 0.3134033

"type" : "eyeRight",
"x" : 0.47972977,
"y" : 0.29185107

"type" : "nose",
"x" : 0.37914225,
"y" : 0.48972052

"type" : "leftPupil",
"x" : 0.2590145,
"y" : 0.30793554

"type" : "rightPupil",
"x" : 0.49394885,
"y" : 0.29406565

"type" : "leftEyeBrowLeft",
"x" : 0.18010405,
"y" : 0.26402563

"type" : "mouthUp",
"x" : 0.38878468,
"y" : 0.5903601

"type" : "mouthDown",
"x" : 0.3908939,
"y" : 0.6531675

"pose" :
"roll" : -4.6593237,
"yaw" : -1.4678677,
"pitch" : -8.147724

"quality" :
"brightness" : 43.485928,
"sharpness" : 99.99881

"confidence" : 99.99999

Detected labels for image
Human: 99.21805
People: 99.21803
Person: 99.21805
Looking for celebrities in image
0 celebrity(s) were recognized.

1 face(s) were unrecognized.
Detected lines and words for image

```

Figur 14. Visar alla svar som är möjliga att få från Amazons tjänst Amazon Rekognition

Till varje attribut i svaret från Amazon tillkommer en siffra mellan 0 och 100 som indikerar på hur säkert det är att ett specifikt objekt som tex en cykel finns med på bilden. För ansiktsdetaljerna följer dessutom ett värde 0 eller 1. 0 indikerar på att ögonen inte är öppna med en viss säkerhet och 1 att de är öppna, se *tabell 1* [12].

Tabell 1 visar parametrar och dess sannolikhet

Eyes Open	0	99.321
Smile	1	89.432

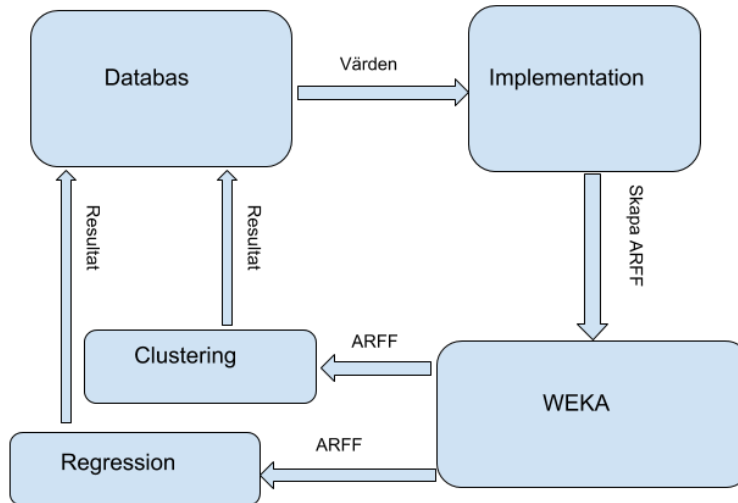
I *tabell 1* avläses första raden som att ögonen är stängda med 99.321% säkerhet 0 indikerar alltså på att ögonen inte är öppna. Ur andra raden avläses att personen ler med 89.432% säkerhet 1 indikerar alltså på att det är ett leende. Den medföljande 0 eller 1 i kolumn två indikerar alltså på om påståendet

i kolumn ett är sant eller falskt. Siffran i kolumn tre är den data som används när instanserna byggs upp. För att undvika att lägga till ett motsatsattribut för varje ansiktsdetalj sparas värdet som 100 - värdet i kolumn tre om värdet i kolumn två är 0.

Alla attribut som på något sätt skulle kunna vara intressanta sparas i databasen men inte alla används i examensarbetet på grund av "Curse Of Dimensionality" se avsnitt 2.2 och 3.5.6.

3.5.2 WEKA OCH ARFF

För att kunna analysera data med Wekas bibliotek måste datan vara inlagd i ett ARFF-dokument. Dokumentet genereras med information från databasen. Dokumentet byggs med de attribut som är intressanta för processen och instanserna. Algoritmerna anropas genom WEKAs API och sparar sedan resultatet i databasen. *Figur 15* visar hur värdena från Amazon Rekognition sparas i databasen för att sedan hämtas och för att skapa ARFF-dokumentet.



Figur 15 Diagrammet visar flödet från lagrade värden till resultat

På de rader i ARFF-dokumentet där “@attribut” står, se *figur 16*, presenteras alla parametrar som kan användas när en inlärningsmetod körs i WEKAs algoritmer. Genom WEKAs API kunde attribut väljas ut så att endast önskade attribut användes vid inläringen.

```

@relation myRelation

@attribute ID numeric
@attribute Beard numeric
@attribute Eyeglasses numeric
@attribute EyesOpen numeric
@attribute MouthOpen numeric
@attribute Mustache numeric
@attribute Smile numeric
@attribute Sunglasses numeric
@attribute Sharpness numeric
@attribute Brightness numeric
@attribute HAPPY numeric
@attribute SAD numeric
@attribute ANGRY numeric
@attribute CONFUSED numeric
@attribute DISGUSTED numeric
@attribute SURPRISED numeric
@attribute CALM numeric
@attribute UNKNOWN numeric

@data
1,1,99,51,1,99,1,99,50,1,99,0,0,1,0,0,0,0
2,60,99,99,82,22,98,1,99,61,99,1,0,6,0,0,0,0
3,1,98,11,99,71,1,100,43,62,99,0,0,0,0,0,1,0
4,3,1,42,2,2,99,1,99,52,97,0,0,18,0,0,0,0
5,18,81,99,96,1,99,55,51,1,93,0,0,0,0,0,16,0
6,1,1,99,19,1,99,1,99,50,99,0,0,0,0,0,0,0
7,97,2,99,57,1,1,95,72,1,99,0,7,0,0,0,6,0
8,1,1,93,1,2,87,1,99,63,95,2,0,0,0,0,15,0
9,45,1,93,1,99,32,1,1,99,99,0,0,1,0,0,0,0
10,1,1,99,53,1,95,1,99,45,99,0,0,1,0,0,1,0
11,1,99,32,22,99,1,99,35,61,99,0,0,0,0,0,0,0
12,45,1,99,94,27,1,92,1,99,99,0,0,2,0,0,0,0
13,0,1,99,12,0,90,1,99,53,95,5,0,0,0,0,13,0
14,1,99,53,1,1,99,10,1,96,99,1,0,0,0,0,1,0

```

Figur 16 visar attribut och instanser för ett dataset på ARFF-format

Varje rad under @data i *figur 16* motsvarar en instans, dvs ett ansikte för en specifik bild. Varje instans bedöms individuellt oavsett om en bild innehåller flera instanser. I de enskilda fall där en bild innehåller fler ansikten än ett, kommer samma bild-id att finnas med på flera ställen i ARFF-dokumentet. Varje ansikte representeras av en instans eftersom att om någon av personerna på bilden blundar eller ser sur ut ska bilden inte väljas ut som en bra bild. Detta kontrollerar dock inte systemet då det bara utvärderar porträtt med en person.

Attributet ID var tvunget att följa med som ett attribut för att kunna följa vilken klass eller vilket kluster en instans hamnat i efter en algoritm körts. Detta var tvunget eftersom att bilder skulle presenteras om de hade klassats som bra bilder eller hamnat i ett specifikt kluster. Det fanns ingen inbyggd "tracking metod" i weka så detta behövde implementeras manuellt. Detta gjordes med hjälp av ID-attributet för en instans.

3.5.3 CLUSTERING

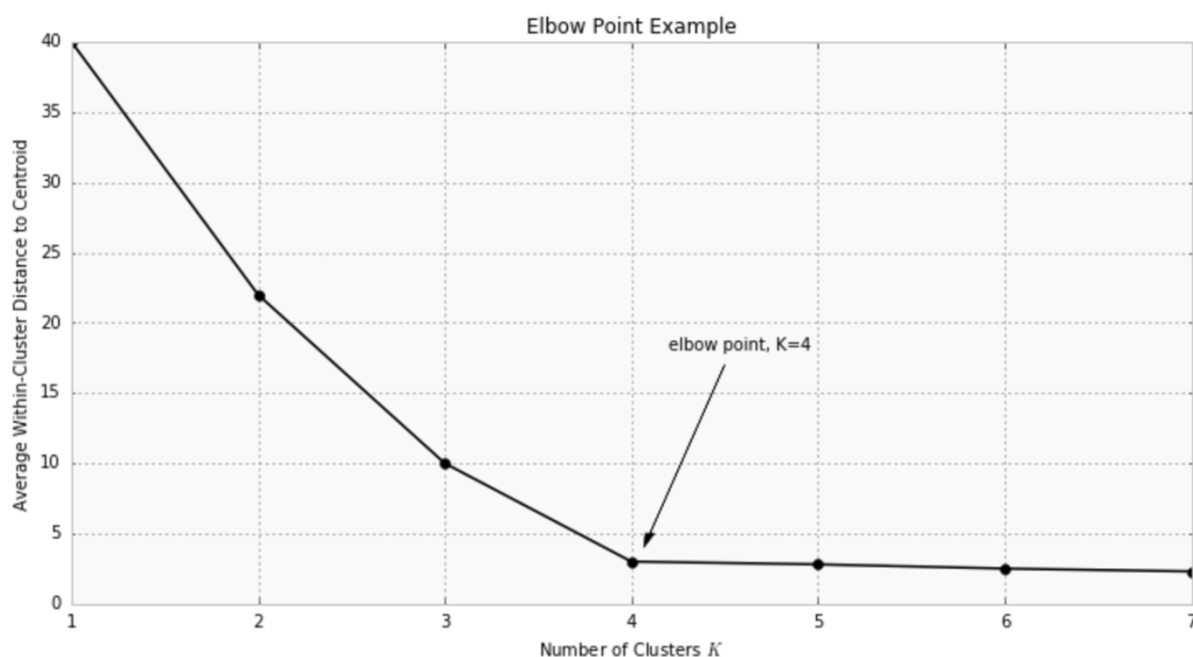
Välja antal Kluster

Valet av antal kluster för en datasamling är viktigt för att få det bästa möjliga resultatet. För att välja det antal kluster som med störst sannolikhet kommer ge det bästa resultatet kan man använda en teknik som räknar ut "Elbow point". Elbow Point refererar till ett specifikt antal kluster.

Det som är intressant vid uträkningen är medeldistansen mellan instanser i ett kluster. Högsta medeldistansen fås vid användning av ett kluster och lägsta, det vill säga 0, fås när antalet kluster är lika med antalet instanser. Det Elbow Point används till är att hitta det kluster där lutningen före och efter skiljer sig mest. Detta visualiseras enklast med hjälp av en graf, *se figur 17* [11]. Lutningen beräknas genom att räkna ut riktningskoefficienten före och efter ett givet kluster det görs med formeln:

$$k = \frac{y_2 - y_1}{x_2 - x_1} \quad [16]$$

Där y är medelavståndet, k lutningen och x antal kluster



Figur 17 visar en graf över Elbow Point [11]

Vid användandet av Elbow Point erhöjls inget tydligt resultat. Lutningen förhöjls sig relativt jämt fördelad i längs hela grafen. Därför användes två kluster under testerna för att gruppera bra bilder för sig och dåliga för sig eftersom att det till synes gav det bästa resultatet för det antal bilder som examensarbetet hade tillgång till.

Välja algoritm för Clustering

Både K-means och Density Based Clustering testades som algoritmer. Den Clusteringalgoritm som valdes var Density Based Clustering eftersom den gav bäst resultat vid inledande tester vilket skulle kunna härledas till att klustren skapas med avseende på densitet, se avsnitt 2.1.2.

3.5.4 LINJÄR REGRESSION

Det fanns inte bara behov av att förutspå värden, utan även att kategorisera bilderna som antingen bra eller dåliga. Den linjära regressionen gav möjlighet till att förutspå värden, men inte att kategorisera. Därför byttes metoden till logistisk regression, som behandlas i nästa avsnitt, se 3.5.5.

3.5.5 LOGISTISK REGRESSION

Eftersom examensarbetets mål var att kunna klassificera instanser så valdes logistisk regression istället för linjär regression. Hur logistisk regression kan klassificera behandlas i avsnitt 2.1.2 om logistisk regression. Valet föll på logistiska regression eftersom det inte var intressant hur bra en bild var utan om den föll in under kategorin “bra bilder” vilket i detta fallet blir den övergripande skillnaden av vad som var möjligt med logistisk regression gentemot linjär regression.

För att träna klassifikatorn skapades ett ARFF-dokument, se avsnitt 3.5.2, där instanserna var kategoriserade. Detta dokumentet läses sedan in och varje instans skapar en datapunkt i funktionen.

Hur väl algoritmen kan klassificera beror till stor del på hur bra träningsdatan representerar hela datamängden. Därför har bra och dåliga bilder noggrant valts ut och lagts in i systemet. För att träna klassifikatorn på vad som är en bra bild valdes porträtt där personen på bilden såg glad ut, log och hade ögonen öppna. För att träna klassifikatorn på dåliga bilder valdes porträtt där personen blundade, gjorde en konstig grimas eller såg sur ut.

3.5.6 ANTALET PARAMETRAR

Att välja rätt parametrar var viktigt för att få önskat resultat, vilket för detta arbete var ett kluster med bra bilder och ett med dåliga bilder. Parametrarna kunde inte ge ett felaktigt resultat, utan visade då endast klustret ur en annan synvinkel vilket gav annan information. Vissa synvinklar blir mer användbara än andra. Utmaningen har legat i att hitta rätt parametrar för att välja ut bra bilder. Det fanns ingen färdig metod som gav svar på vilka parametrar som borde användas för bäst resultat. Genom att testa ett antal olika parametrar valdes HAPPY och Smile eftersom de bäst gav den önskade synvinkeln.

Via Amazon Rekognition gavs tillgång till en mängd olika parametrar. Ett kriterium för att en bild skulle klassas som bra, var att personen på bilden såg glad ut. Det fanns flera parametrar från Amazon Rekognition som kunde uppfylla detta; eyesOpened, mouthOpened och smile. Dessutom var det även möjligt att få ut personens känslomässiga uttryck som exempelvis HAPPY, SAD och CALM.

Enligt Curse Of Dimensionality (avsnitt 2.2) var det endast möjligt att fokusera på maximalt två parametrar. Olika parametrar testades och resulterade i insikten att parametern HAPPY var en fördelaktig parameter. Den kunde sammanfatta flera parametrar, som exempelvis mouth opened och

eyesopend. För att förstärka effekten ytterligare och ge ett smalare urval av bilder adderades en parameter till, smile. Detta ledde till att metoderna testades med parametern HAPPY, men också med parametrarna HAPPY och Smile. 3.5.7 Förhållande mellan parametrar och bilder

Mängden data är relationell till mängden parametrar. Det innebär att fler parametrar kräver en större mängd data. Algoritmer som Clustering försöker hitta samband mellan de olika bildernas attribut. Om det finns för många parametrar i förhållande till mängden data blir skillnaden mellan olika kluster otydlig.

Examensarbetet behövde ta hänsyn till Curse of Dimensionality eftersom att det inte fanns färdiga dataset att använda. Istället behövde dessa skapas under examensarbetet vilket var tidskrävande. Förhållandet mellan mängd data och antal parametrar ökar exponentiellt. Därför testades 1 parameter på 50 bilder och 2 parametrar på 500 bilder, se avsnitt 2.2.

3.6 ANVÄNDARTESTER

Användartester genomfördes enskilt på några utav New Seeds anställda, där varje testperson valt ut ett antal bilder som systemet skulle reagera på. Testerna gjordes på två system, ett som använde Clustering och ett som använde Regression för att presentera bra bilder.

Testmiljön var uppbyggd av två exekverbara program i Eclipse, ett program för vardera maskininlärningsmetod. Egenskrivna metoder i Eclipse användes för att klassificera och analysera testdatan. Resultatet samlades och presenterades i HTML-filer som visades för testpersonen.

3.7 VERKTYG

Här beskrivs de verktyg som använts under examensarbetet.

3.7.1 EUCLIDEAN DISTANCE

Euclidean Distance är en algoritm som används för att beräkna avstånd mellan datapunkter inom metoderna; Regression och Klassificering. Nedanstående formel används för att beräkna avståndet(a) för varje dimension:

$$a = \sqrt{w_1^2(x_1 - y_1)^2 + w_2^2(x_2 - y_2)^2 + \dots + w_n^2(x_n - y_n)^2} [5]$$

För att beräkna avståndet med avseende på alla vikter för varje attribut för varje dimension. Alla attributens vikter uppdateras när träningsinstanserna blir klassificerade. De mest lika instanserna används som bas vid nästa beräkning. För träningsinstansen (x), dess mest lika instans (y) och attribut(i) beräknas skillnaden för varje attribut $x_i - y_i$. Skillnaden blir bidraget som avgör hur vikten ska justeras. Vid liten skillnad får bidraget positivt värde och vid stor skillnad får bidraget negativt värde. Varje attributs vikt kommer att uppdateras baserat på storleken på bidraget och om klassificeringen var korrekt. Om klassificeringen är korrekt kommer vikten att ökas och om klassificeringen är inkorrekt kommer vikten att minskas. Mängden ökning och minskning avgörs av storleken på bidraget.

3.7.2 GIT

Git har använts som versionshanterare och för att dela kod. Bitbucket har använts som globalt Repository och sourceTree som lokal Repository.

3.7.3 WEKA

WEKA är ett bibliotek med maskininlärning- och datasamlingsalgoritmer som antingen kan användas genom ett interface eller anropas från JAVA-kod. WEKA innehåller klasser och metoder som gör det möjligt att analysera data genom bland annat Clustering, klassificering och Regression. WEKA är en open source-mjukvara [13]. Examensarbetet har inte använt sig av WEKA interface, utan enbart de klasser och metoder som WEKA tillhandahåller [14].

3.7.4 AMAZON REKOGNITION

Amazon Rekognition är en tjänst för bild-och videoanalys producerat av Amazon. Tjänsten för bildanalys kan bland annat detektera föremål, personer och ansikten. Tjänsten kan vid en porträttbild detektera ansiktsuttryck, positionering för ögon och om det finns föremål i ansiktet samt ett åldersintervall för personen på bilden. *Figur 18* visar exempel över de värden som är möjliga att få ut vid analys [12].

The detected face is estimated to be between 19 and 36 years old.
Here's the complete set of attributes:

```

"boundingBox"
  "width" : 0.58625,
  "height" : 0.7816667,
  "left" : 0.086875,
  "top" : -0.0125

"ageRange"
  "low" : 19,
  "high" : 36

"smile" :
  "value" : true,
  "confidence" : 85.668655

"eyeglasses" :
  "value" : false,
  "confidence" : 95.25664

"sunglasses" :
  "value" : false,
  "confidence" : 98.98994

"gender" :
  "value" : "Male",
  "confidence" : 99.631226

"beard" :
  "value" : true,
  "confidence" : 61.823742

```

```

"emotions" :
  "type" : "HAPPY",
  "confidence" : 93.65743

  "type" : "CONFUSED",
  "confidence" : 17.817127

  "type" : "ANGRY",
  "confidence" : 0.67237616

"landmarks" :
  "type" : "eyeLeft",
  "x" : 0.26456237,
  "y" : 0.3134033

  "type" : "eyeRight",
  "x" : 0.47972977,
  "y" : 0.29185107

  "type" : "nose",
  "x" : 0.37914225,
  "y" : 0.48972052

  "type" : "leftPupil",
  "x" : 0.2590145,
  "y" : 0.30793554

  "type" : "rightPupil",
  "x" : 0.49394885,
  "y" : 0.29406565

  "type" : "leftEyeBrowLeft",
  "x" : 0.18010405,
  "y" : 0.26402563

```

```

"type" : "mouthUp",
"x" : 0.38878468,
"y" : 0.5903601

"type" : "mouthDown",
"x" : 0.3908939,
"y" : 0.6531675

"pose" :
  "roll" : -4.6593237,
  "yaw" : -1.4678677,
  "pitch" : -8.147724

"quality" :
  "brightness" : 43.485928,
  "sharpness" : 99.99881

"confidence" : 99.99999

```

Detected labels for image
Human: 99.21805
People: 99.21803
Person: 99.21805
Looking for celebrities in image
0 celebrity(s) were recognized.
1 face(s) were unrecognized.
Detected lines and words for image

Figur 18 visar värden som fås ut vid analys med Amazon Rekognition

3.7.5 MySQL

MySQL har använts för att spara information om varje bild för att slippa skicka bilder på analys varje gång.

3.8 KÄLLKRITIK

Detta avsnitt behandlar källkritik. Källkritiken behandlar trovärdigheten hos de tre verktyg som examensarbetet använt sig av.

3.8.1 WEKA

Informationen om WEKA är hämtad från University of Waikato. WEKA är programvara skriven med öppen källkod och därför kan påstådd funktionalitet verifieras genom att läsa källkoden.

3.8.2 AMAZON REKOGNITION

Funktionaliteten som Amazon påstår sig ha kan anses som trovärdig då det används dagligen av miljontals människor.

3.8.3 MASKININLÄRNING

Information om hur maskininlärningsalgoritmerna som används av examensarbetet fungerar är hämtad från flertalet olika källor som bekräftar varandra.

3.8.4 VAL AV TRÄNINGSDATA

Den totala mängden träningsdata består av 500 bilder. Hälften av dessa är bra och hälften av dessa är dåliga. Under examensarbetets gång fanns ej tillgång till någon bildbank, istället har privata bilder använts men också bilder från Google och Facebook. Bilderna är valda utefter egna preferenser med hänsyn till de parametrar som valts att fokusera på. Reservation lämnas därför för en möjligt mindre optimal och objektiv träningsdata.

4 RESULTAT

Vid testning av Clustering användes två kluster, ett för att representera bra bilder och ett för att representera dåliga bilder. Detta för att kunna jämföra urvalet mot logistiska Regressionen, som klassificerar utdata i två grupper. Testpersonerna hade med sig egna bilder, där de själva hade bestämt vilka av dessa bilder de ansåg vara bra bilder och vilka de ansåg vara dåliga bilder.

4.1 ANVÄNDARTESTER

Varje testperson hade med sig 8-10 bilder, varav hälften bra och hälften dåliga. Bilderna lades in i systemet och testerna utfördes på träningsdata innehållande 50 eller 500 bilder. Systemet testades på förmåga att kunna välja ut bra bilder baserat på den mängd träningsdata som systemet tränats på. Resultatet från maskininläringen jämfördes med slumpmässigt utvalda bilder, där både antalet utvalda bilder och bilderna slumpades fram. Bilderna som slumpades fram kom från samma mängd testdata som övriga tester genomfördes på. I tabellerna är samtliga bilder märkta med antingen "yes" eller "no", vilka avser bra eller dåliga bilder.

4.1.1 ANVÄNDARTEST EFTER 50 BILDER

Användartester genomfördes på datamängd bestående av 50 bilder, varav 25 bra bilder och 25 dåliga bilder. Testerna genomfördes för att se skillnaden mellan Clustering och Logistisk Regression. För båda metoderna gjordes även tester med en parameter respektive två parametrar.

Clustering med en parameter

Clustering testades med parametern HAPPY på bilderna som testpersonerna hade med sig. Ur *tabell 3* avläses att kluster 0 innehåller 37 bilder och att kluster 1 innehåller 13 bilder. Ur *tabell 3* går även att avläsa centroiden för varje attribut. Centroiden i kluster 0 fanns på 95,5676 och på 12,1538 för kluster 1, båda för attributet HAPPY. Testet visade att kluster 0 var det bästa klustret, eftersom parametern HAPPY hade högst medelvärde.

Ur *tabell 2* avläses resultaten för testerna. Det går att avläsa andelen testbilder som ansågs bra respektive dåliga för varje test, samt fördelningen mellan bra och dåliga bilder för olika kluster. Resultaten för testen med slumpvist utvalda bilder ur bilderna som testpersonerna hade med sig finns även i *tabell 2*.

Av testbilderna valdes totalt 21 bra och 7 dåliga bilder ut som bra bilder och hamnade i kluster 0. Av de slumpmässigt utvalda testbilderna valdes totalt 9 bra och 10 dåliga bilder ut.

Tabell 2 visar resultatet över bilder utvalda av Clustering med parametern HAPPY

Test	Testbilder		Utvalda Clustering				Utvalda Slumpmässigt	
	Yes	No	Kluster 0		Kluster 1		Yes	No
			Yes	No	Yes	No		
1	3	6	3	2	0	4	1	1
2	5	5	5	1	0	4	0	0
3	4	5	4	2	0	3	3	3
4	5	5	5	2	0	3	3	5
5	5	5	4	0	1	5	2	1
Summa	22	26	21	7	1	19	9	10

Tabell 3 visar fördelning av bilder mellan olika kluster

Attribut	Full data (50,0)	Kluster 0 (37)	Kluster 1 (13)
Happy	73,55	95,5676	12,1538

Clustering med två parametrar

Clustering med parametrarna HAPPY och Smile utfördes på bilderna som testpersonerna hade med sig. Ur *tabell 5* avläses att kluster 0 innehåller 37 bilder och att kluster 1 innehåller 13 bilder. I kluster 0 för attributet Smile fanns centroiden på 94,5833 och för kluster 1 på 22,2143. För attributet HAPPY fanns centroiden för kluster 0 på 96,1389 och för kluster 1 på 16,6429. I detta test var kluster 0 det bästa klostret, eftersom både parametrarna HAPPY och Smile hade högst medelvärde.

Ur *tabell 4* avläses resultaten för testerna. Det går att avläsa andelen bilder som ansågs bra respektive dåliga för varje test, samt fördelningen mellan bra och dåliga bilder i olika kluster. Resultaten för testen med slumpmässigt utvalda bilder från testdatan finns även i *tabell 4*.

Av testdatan hamnade totalt 20 bra och 6 dåliga bilder i kluster 0, jämfört med de slumpmässigt utvalda bilderna som innehåller totalt 9 bra och 10 dåliga bilder.

Tabell 4 visar resultatet över bilder utvalda av Clustering med parametern Happy och Smile

Test	Testbilder		Utvalda Clustering				Utvalda Slumpmässigt	
	Yes	No	Kluster 0		Kluster 1		Yes	No
			Yes	No	Yes	No		
1	3	6	3	2	0	4	2	5
2	5	5	5	1	0	4	2	1
3	4	5	4	2	0	3	0	0
4	5	5	5	1	0	4	3	3
5	5	5	3	0	2	5	1	1
Summa	22	26	20	6	2	20	8	10

Tabell 5 visar fördelning av bilder mellan olika kluster.

Attribut	Full data (50)	Kluster 0 (36)	Kluster 1 (14)
Smile	74,32	94,5833	22,2143
Happy	73,88	96,1389	16,6429

Logistisk Regression med en parameter

Logistisk Regression med parametern HAPPY utfördes på testdatan där klassifikatorn tränats med 50 instanser. Ur *tabell 6* avläses att logistisk Regression klassar totalt 17 bra och 3 dåliga bilder som bra bilder. Av de slumpmässigt utvalda bilderna är totalt 7 bra och 5 dåliga [10].

Tabell 6 visar resultatet över bilder utvalda av Regression med parametern Happy

Test	Testbilder		Klassificerade som bra		Utvalda Slumpmässigt	
	Yes	No	Yes	No	Yes	No
1	3	6	3	2	2	1
2	5	5	3	0	1	1
3	4	5	4	1	1	1
4	5	5	3	0	3	2
5	5	5	4	0	0	0
Summa	22	26	17	3	7	5

Logistisk Regression med två parametrar

Logistisk Regression med parametern Happy utfördes på testdatan där klassifikatorn tränats med 50 instanser. Ur *tabell 7* avläses att logistisk Regression klassar totalt 12 bra och 4 dåliga bilder som bra bilder. Av de slumpmässigt utvalda bilderna är totalt 14 bra och 14 dåliga [10].

Tabell 7 visar resultatet över bilder utvalda av Regression med parametrarna Happy och Smile

Test	Testbilder		Klassificerade som bra		Utvalda Slumpmässigt	
	Yes	No	Yes	No	Yes	No
1	3	6	3	2	2	5
2	5	5	2	0	2	3
3	4	5	4	2	2	1
4	5	5	3	0	4	3
5	5	5	0	0	4	2
Summa	22	26	12	4	14	14

Skillnad i resultat mellan metoder vid 50 bilder

Efter test med träningsdata på 50 bilder syntes att Clustering valde ut fler bilder som bra och fler dåliga bilder som bra, än vad Regressionen gjorde. Detta gjordes för både en och två parametrar. Vid användning av Regression valdes färre bra bilder ut som bra, samtidigt som färre dåliga bilder valdes ut som bra. Se *diagram 1* för sammanställning.

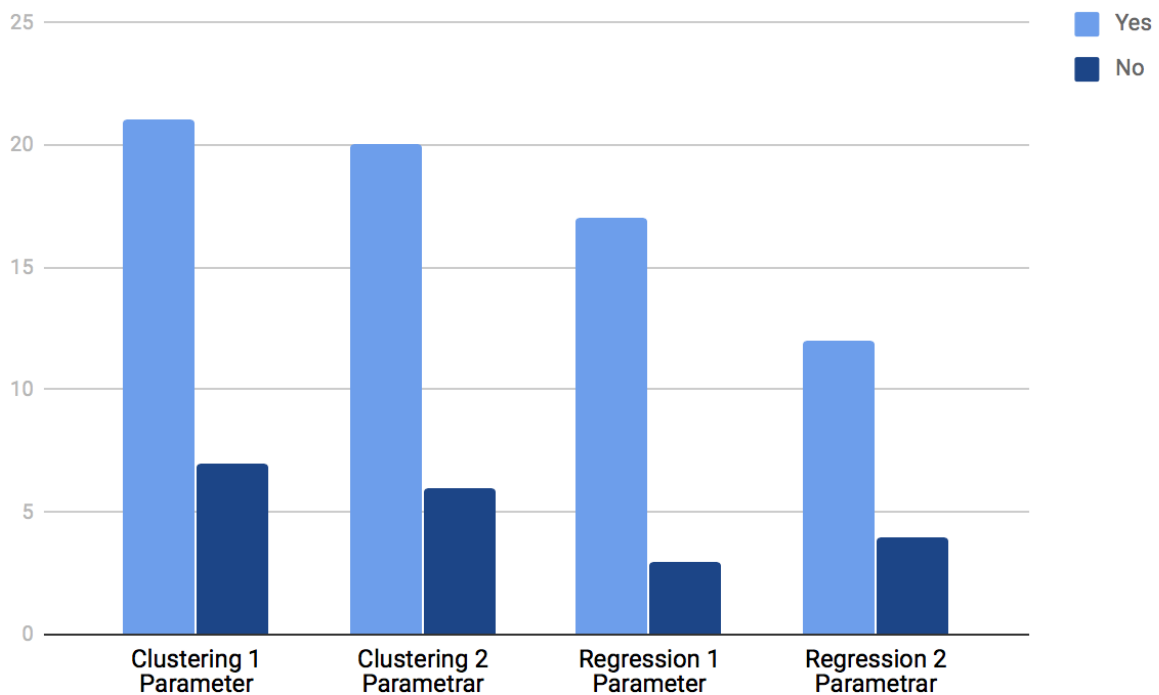


Diagram 1 visar antalet bra och dåliga bilder för varje metod vid träningsdata på 50 bilder.

4.1.2 ANVÄNDARTEST EFTER 500 BILDER

Clustering med en parameter

Clustering testades med parametern HAPPY på bilderna som testpersonerna hade med sig. Ur *tabell 9* avläses att kluster 0 innehåller 318 bilder och kluster 1 innehåller 167 bilder. För attributet HAPPY fanns centroiden för kluster 0 på 94,4245 och för kluster 1 på 13,479. I detta test var kluster 0 det bästa klustret, eftersom värdet för parametern HAPPY hade högst medelvärde.

Ur *tabell 8* avläses resultaten för testerna. Det går att avläsa andelen testbilder som ansågs bra respektive dåliga för varje test, samt fördelningen mellan bra och dåliga bilder för olika kluster. Även resultaten från testerna med slumpvis utvalda bilder ur testdatan finns i *tabell 8*.

Av testdatan valdes totalt 17 bra bilder och 8 dåliga som bra bilder och hamnade i kluster 0. Av de slumpmässigt utvalda bilderna valdes totalt 10 bra och 14 dåliga bilder ut [10].

Tabell 8 visar resultatet över bilder utvalda av Clustering med parametern HAPPY

Test	Testbilder		Utvalda Clustering				Utvalda Slumpmässigt	
	Yes	No	Kluster 0		Kluster 1		Yes	No
			Yes	No	Yes	No		
1	3	6	3	2	0	4	1	3
2	5	5	5	1	0	4	5	4
3	4	5	0	3	4	2	2	3
4	5	5	5	2	0	3	2	2
5	5	5	4	0	1	5	0	2
Summa	22	26	17	8	5	18	10	14

Tabell 9 visar fördelning av bilder mellan olika kluster

Attribut	Full data (485)	Kluster 0 (318)	Kluster 1 (167)
Happy	66,5526	94,4245	13,479

Clustering med två parametrar

Clustering med parametrarna HAPPY och Smile utfördes på bilderna som testpersonerna hade med sig. Ur *tabell 11* avläses att kluster 0 innehåller 164 bilder och kluster 1 innehåller 321 bilder. I kluster 0 för attributet Smile fanns centroiden på 23,0488 och för kluster 1 på 93,0592. För attributet HAPPY fanns centroiden för kluster 0 på 15,1098 och för kluster 1 på 93,134. I detta test var kluster 1 det bästa klustret, eftersom både Smile och HAPPY hade högst medelvärde.

Ur *tabell 10* avläses resultaten för testerna. Det går att avläsa andelen bilder som ansågs bra respektive dåliga för varje test, samt fördelningen mellan bra och dåliga bilder i olika kluster. Resultaten för testen med slumpmässigt utvalda bilder finns även i *tabell 10*.

Av testdatan hamnade totalt 20 bra bilder och 6 dåliga i kluster 1, jämfört med de slumpmässigt utvalda bilderna som innehåller totalt 9 bra och 10 dåliga bilder.

Tabell 10 visar resultatet över bilder utvalda av Clustering med parametern HAPPY och Smile

Test	Testbilder		Utvalda Clustering				Utvalda Slumpmässigt	
	Yes	No	Kluster 0		Kluster 1		Yes	No
			Yes	No	Yes	No		
1	3	6	0	3	3	3	1	3
2	5	5	0	4	5	1	5	4
3	4	5	0	2	4	3	2	3
4	5	5	0	4	5	1	2	2
5	5	5	0	5	5	0	0	2
Summa	22	26	0	18	22	8	10	14

Tabell 11 visar fördelning av bilder mellan olika kluster

Attribut	Full data (485)	Kluster 0 (164)	Kluster 1 (321)
Smile	69,3856	23,0488	93,0592
Happy	66,7505	15,1098	93,134

Logistisk Regression med en parameter

Logistisk Regression med parametern HAPPY utförs på testdatan. Ur *tabell 12* avläses att logistisk Regression klassificerar totalt 21 bra bilder och 6 dåliga som bra bilder. Av de slumpmässigt utvalda bilderna är totalt 10 bra och 15 dåliga.

Tabell 12 visar resultatet över bilder utvalda av Regression med parametern Happy

Test	Testbilder		Klassificerade som bra		Utvalda Slumpmässigt	
	Yes	No	Yes	No	Yes	No
1	3	6	3	2	2	5
2	5	5	5	1	5	4
3	4	5	4	1	2	1
4	5	5	5	2	0	1
5	5	5	4	0	1	4
Summa	22	26	21	6	10	15

Logistisk Regression med två parametrar

Logistisk Regression med parametrarna HAPPY och Smile utförs på testdatan. Ur *Tabell 14* avläses att av de bilder som logistisk Regression har valt ut som bra bilder, är totalt 20 bra och 7 dåliga. Av de bilder som slumpmässigt valts ut som bra bilder är totalt 13 bra och 11 dåliga.

Tabell 13 visar resultatet över bilder utvalda av Regression med parametrarna Happy och Smile

Test	Testbilder		Klassificerade som bra		Utvalda Slumpmässigt	
	Yes	No	Yes	No	Yes	No
1	3	6	3	3	0	0
2	5	5	4	1	4	5
3	4	5	4	3	5	1
4	5	5	5	0	1	4
5	5	5	4	0	3	1
Summa	22	26	20	7	13	11

Skillnad i resultat mellan metoder vid 500 bilder

Resultaten för en respektive två parametrar skiljde sig inte nämnvärt. *Diagram 2* visar skillnaden i resultat för metoder och antal parametrar. För representation av Clustering har de bilder som klassades som bra respektive de som klassades som dåliga, hämtats från det bästa klustret.

Ur *diagram 2* går att avläsa att testet på Clustering med en parameter valde ut färre bra bilder och fler dåliga bilder, jämfört med de andra testen. Clustering med två parametrar valde ut 22 bra bilder och 8 dåliga bilder som bra. Skillnaden i prestation för regressionen skiljer sig lite.

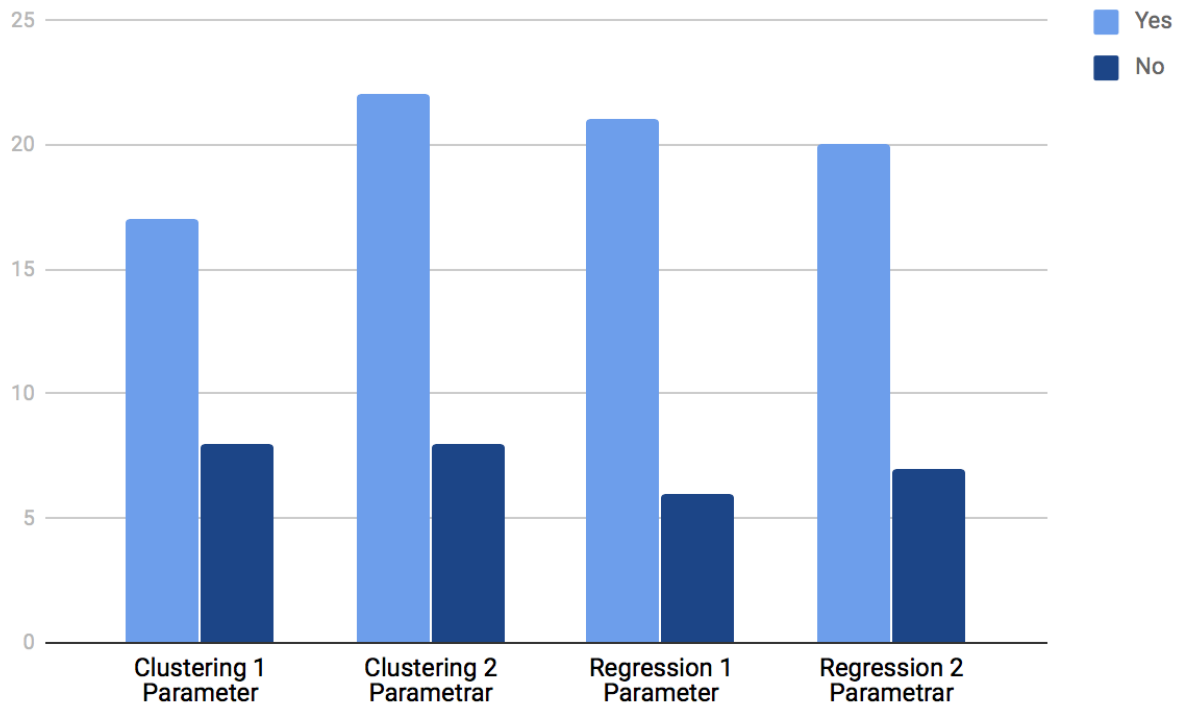


Diagram 2 visar antalet bra och dåliga bilder för varje metod vid träningsdata på 500 bilder

4.2 SAMMANSTÄLLNING AV RESULTAT

För att kunna jämföra metoderna har resultatet från samtliga tester sammanställts i en tabell, se *tabell 14*. För testerna av Clustering har resultatet från det bästa klustret valts och jämförts med resultatet av de slumpmässigt utvalda bilderna. Tabellen visar andelen bra och dåliga bilder som varje metod har valt ut.

En sammanslagning av resultatet för varje metod gjordes genom att ta andelen bra bilder som systemet valde ut, dividerat med den totala mängden bra bilder. För att få fram andelen dåliga bilder som systemet valt ut togs antalet dåliga bilder och dividerat med den totala mängden dåliga bilder.

Sammanläggningen av resultatet från testning av Clustering, med en parameter och 50 bilder i träningsdata, visar att 95% av de bilder som Clustering valde ut är bra bilder, enligt testpersonerna. Sammanläggningen av samma test visar också att 27% av de bilder som testpersoner ansåg vara dåliga, valdes ut som bra bilder av systemet. Övriga 5 % av de bilder som testpersonerna ansåg som

bra, valdes av systemet ut som dåliga. De resterande 73 % av bilderna som testpersonerna ansåg vara dåliga, valdes ut av systemet som dåliga bilder.

Tabell 14 visar andelen utvalda bilder för respektive metod

Metod	Antal parametrar	Mängd träningsdata	Utvalda		Slump	
			Yes	No	Yes	No
Clustering	1	50	95%	27%	41%	38%
Clustering	2	50	90%	23%	36%	38%
Regression	1	50	77%	12%	32%	19%
Regression	2	50	55%	15%	64%	54%
Clustering	1	500	77%	31%	45%	54%
Clustering	2	500	100%	31%	41%	42%
Regression	1	500	95%	23%	45%	58%
Regression	2	500	91%	27%	59%	42%

5 DISKUSSION OCH ANALYS

Som framgår under avsnitt 4 var det möjligt att använda Clustering så väl som Regression för att i högre utsträckning välja ut bra bilder än när samma bilder väljs ut slumpmässigt. När procentsnittet räknas ut på bra respektive dåliga bilder för alla tester visar det sig att 85% av de bilder som klassificerades som bra presenterades till skillnad från 23.625% av de dåliga. Detta ger en tydlig indikering på att de mål som sattes upp för examensarbetet har uppnåtts.

Om man ser till hur resultaten förändras med avseende på hur många bilder som använts vid träning ser man att Clustering inte förbättrats vid användning av 1 parameter vid träning med 50 gentemot 500 bilder, utan dessvärre försämrats. Däremot går det att ana en förbättring vid användandet av 2 parametrar. När man kontrollerar samma förändring vid användandet av logistisk Regression blir resultaten avsevärt bättre speciellt vid användandet av 2 parametrar då andelen framtagna bilder som är bra ökar från 55% till 91%.

En anledning till att Clustering inte förbättrades mer skulle kunna vara att antalet kluster inte ökades i samband med att fler bilder lades till. Valet att bara använda sig av två kluster ansågs rimligt eftersom att problemet var att klassificera bilder i 2 grupper. Vid analys av detta hade examensarbetet möjligtvis nått bättre resultat om fler kluster hade använts. Detta hade dessvärre lett till en något mer komplex implementation och hade inte hunnits med sett i efterhand. Men i ett vidareutvecklingsscenario borde det testas och utvärderas.

I ett slutanvändningsscenario kan man anse att det inte gör så mycket om en mindre del dåliga bilder väljs ut eftersom att användaren lätt sorterar ut de, ett sämre fall skulle antagligen vara om man får ett resultat liknande det i tabell 7. Där väljs relativt få bra bilder ut sett till de andra resultaten, men också relativt få dåliga. Detta skulle kunna ses som ett sämre resultat eftersom att användaren då går miste om 45% av de bra bilderna i bildsamlingen.

Många saker har inverkan på vilka bilder som väljs ut, en anledning till att sämre bilder väljs ut bland de bra är att Amazon inte levererar korrekta analyser. Under testning visade det sig att ansikten som en människa uppfattar som likgiltiga eller rent av ledset kan Amazon läsa av och beskriva som "HAPPY" med hög säkerhet. Detta inträffade vid analys av en bild, där det ansågs att personen på bilden såg ledsen ut. Den ledsna munnen valde Amazon att tolka som ett leende, vilket resulterade i att bilden klassades som "HAPPY" med hög sannolikhet.

Systemet som implementerats i detta examensarbete utför inte någon kontroll på om parametrarna som Amazon levererar är korrekta, därför har Amazon Rekognition som tredjepartsverktyg stor inverkan på resultatet. Vilken träningsdata som använts och hur många instanser som algoritmerna tränats på har också tydlig påverkan vilket märktes när träningsdata alternerades med samma mängd bilder. För träningsdata är det viktigt att den så bra som möjligt avspeglar den data som ska klassificeras. Om en bild som ska klassificeras inte liknar de bilder som använts vid träningen kommer algoritmen ha svårare att klassificera den på ett korrekt sätt.

När användartesterna gjordes fanns inte tillräckligt med restriktioner på vilken typ av bilder systemet klarar av att klassificera. Exempelvis tog inte systemet hänsyn till om bilden var mörk eller suddig vilket vissa av testbilderna var som var tänkta som dåliga. Mängden träningsdata påverkar också resultatet, ju större mängd träningsdata som använts ju högre precision. Dessutom kan man med en större mängd träningsdata använda fler attribut att klassificera på vilket i sin tur ger möjlighet att klassificera ett bredare urval av bilder.

Tillvägagångssättet att bygga upp dataset på det vis som beskrivs i avsnitt 3.5.2 kan möjligtvis ha påverkat resultatet. Om det istället byggts med två attribut för varje detalj det vill säga ett för öppna ögon och ett för stängda ögon hade kunnat vara fördelaktigt då "negativa" attribut hade kunnat skilja bilderna åt. Detta hade dessvärre tvingat oss att använda fler attribut och därmed orsakat "curse of dimensionality" som beskrivs i avsnitt 2.2

Ett sätt att sälla bort sämre svar från amazon skulle kunna vara att bara ta med de attribut där säkerheten är över 90%. När säkerheten kommer under det så kan man tydligt se sämre precision. Istället hade man kunnat sätta den till 0 eller "missing value" för att möjligtvis få bättre resultat.

6 SLUTSATS

Syftet med examensarbetet var att ta fram ett verktyg som ur en bildsamling med en godtycklig mängd bilder kan välja ut de bilder som upplevs bra. Detta skulle ske med hjälp av maskininlärning och uppbyggnad av en testmiljö.

För att avgöra om målet uppfyllts utformades användartester eftersom bedömningen av om en bild är bra är subjektiv behövdes testerna göras på personer.

6.1 VILKA UTAV METODERNA REGRESSION OCH CLUSTERING KAN BÄST FÅ SYSTEMET ATT VÄLJA UT BRA BILDER?

Logistisk Regression lärde sig bäst, då det var störst skillnad på resultat vid en större mängd träningsdata jämfört med en mindre mängd träningsdata. Clustering påverkades mer av antalet parametrar i förhållande till mängd träningsdata, än den faktiska mängden träningsdata. Båda metoderna väljer ut bra bilder med högre sannolikhet än när bilder väljs ut slumpvist. Vid större mängder träningsdata dras slutsatsen att Logistiska Regressionen kommer vara den metod som presterar bäst. Det pekar på att den Logistiska Regressionen är mer lämpad för denna typ av projekt.

6.2 VILKA ATTRIBUT GAV BÄST RESULTAT?

Vilka attribut som gav bäst resultat varierade beroende på mängden data, antal parametrar och metod. Clustering hade bäst resultat vid de två parametrarna happy och smile, tillsammans med den större mängden träningsdata. Regressionen presterade som bäst vid den större mängden träningsdata med enbart parametern happy.

6.3 HUR MÅNGA PARAMETRAR BEHÖVDE ANVÄNDAS FÖR ATT FÅ BÄST RESULTAT?

För den Logistiska Regressionen spelade mängden träningsdata större roll, än antalet parametrar. Där sågs en tydlig ökning av precision i förhållande till mängd träningsdata. Resultatet för Clustering varierade i förhållande till mängd träningsdata och antal parametrar. För att få bra resultat med Clustering krävs noggrannhet och följsamhet till Curse Of Dimensionality.

6.4 REFLEKTION ÖVER ETISKA ASPEKTER

Detta kapitel behandlar de etiska aspekter som uppkommit under examensarbetets gång.

6.4.1 SAMHÄLLSNYTTA

Systemet bidrar till en effektivisering av e-handeln och därigenom gynnas ekonomin vilket i sin tur gynnar samhället. Genom att effektivt och precist kunna gallra ut oönskade bilder kan flera branscher effektiviseras. Specifikt branscher där många bilder generas som exempelvis branscher som arbetar övervakning eller fotografering men möjligtvis också branscher som detta systemet skapats för.

Exempelvis om ett system liknande det som skapats under examensarbetet skulle kunna välja ut ett bra porträtt för ett CV eller personligt brev så att chansen till en anställning ökar skulle det kunna hjälpa arbetslösa till en snabbare anställning.

6.4.2 ETISKA DILEMMAN

När man låter ett datorsystem avgöra vilka bilder som är bra kommer samhällsnormen bli starkare. Detta är en etisk aspekt som borde diskuteras närmare om ett system ska användas i detta syfte. Ökade normer kan anses ohälsosamt för ett samhälle och därav borde system inte användas med syfte att välja ut bilder som de flesta anser som bra.

6.5 FRAMTIDA UTVECKLINGSMÖJLIGHETER

Under examensarbetets gång har det funnits tankar kring vad en vidareutveckling av systemet kan användas till. Ett förslag skulle kunna vara att låta fotografer använda det för att slippa gå igenom stora bildsamlingar. Ett annat förslag skulle kunna vara att använda systemet för att avgöra om en bild är lämplig att använda som passfoto.

6.5.1 FÖRBÄTTRING

Mer noggrant utvärdera de brister som finns i Amazon Rekognition och förebygga dessa. Tankar till vad detta kan vara för brister tas upp i diskussionen.

6.5.2 ANVÄNDNINGSSOMRÅDEN

Ett användningsområde skulle kunna vara att låta fotografer använda systemet för att slippa gå igenom stora samlingar bilder. Då måste systemet vara mycket mer precist och använda positioner för

ansiktsdetaljer se avsnitt 3.5.1. Med tanke på hur många bilder en fotograf tar under en fotografering idag skulle ett välfungerande verktyg vara både till god hjälp och tidsbesparande.

Smala områden

Områden som ett liknande system kan användas i måste ha tydliga riktlinjer om det ska användas professionellt eller i industrin. Ett användningsområde skulle kunna vara att använda det för att kontrollera om ett passfoto uppfyller de kriterier som ställs på ett sådant. Då hade man behövt jobba mer med positionering av ansiktsdetaljer vilket går att göra med Amazon Rekognition. Detta hade kunnat vara ett tillräckligt smalt område men om de bilder som behövs finns tillgängliga utforskas vidare.

7 KÄLLFÖRTECKNING

- [1] Mitchell, Tom. 1997. *Machine Learning*. New York City: McGraw-Hill
- [2] Russell. Stuart J., Norvig. P. 2003, *Artificial Intelligence - A Modern Approach*. 2.uppl Upper Saddle River, N.J : Prentice Hall
- [3] Russell. Stuart J., Norvig. P. 2010 *Artificial Intelligence - A Modern Approach*. 3.uppl Boston : Pearson Education
- [4] Aurélien Geron. 2017. *Hands-On Machine Learning with Scikit-Learn and TensorFlow*. Sebastiaanpool, Kalifornien : o'Reilly Media
- [5] Alpaydin, Ethem. 2010. *Introduction to Machine Learning*. 2.uppl Cambridge, Massachusetts : The MIT Press
- [6] Ian H. Witten, Eibe Frank, Mark A. Hall. 2016. *Data mining Practical Machine Learning Tools and Techniques*. 4.uppl. Morgan Kaufmann
- [7] Johan Andersson, Hannes Johansson. 2015. *Using Clustering in a Cognitive Tutor to Identify Mathematical Misconceptions*. Lunds universitet.
- [8] Keogh E., Mueen A. (2017) *Curse of Dimensionality*. In: Sammut C., Webb G.I. (eds) *Encyclopedia of Machine Learning and Data Mining*. Springer, Boston, MA
- [9] PennState, Eberly College of Science. 2018
<https://newonlinecourses.science.psu.edu/stat501/node/252/>
Hämtad (2018-05-15)
- [10] Institute for Digital Research and Education <https://stats.idre.ucla.edu/other/mult-pkg/faq/general/faq-how-do-i-interpret-odds-ratios-in-logistic-Regression/>
Hämtad (2018-05-15)
- [11] Andrea Trevino 12.06.16. *Introduction to K-means Clustering*
<https://www.datascience.com/blog/k-means-Clustering>
Hämtad (2018-05-15)

- [12] Amazon Rekognition. *What is Amazon Rekognition*.
<https://docs.aws.amazon.com/rekognition/latest/dg/what-is.html>
Hämtad (2018-05-15)
- [13] WEKA, Machine Learning Group at the University of Waikato
<https://www.cs.waikato.ac.nz/ml/weka/>
Hämtad (2018-05-15)
- [14] WEKA, Dokumentation. <http://weka.sourceforge.net/doc.dev/overview-summary.html>
Hämtad (2018-02-04)
- [15] *Curse of Dimensionality Two - Georgia Tech - Machine Learning*.
<https://www.Youtube.com/watch?v=OyPcbeiwps8>
Hämtad (2018-05-22)
- [16] Jonas M, Patrik N. 2011. *Endimensionell Analys*. Lund :
Studentlitteratur
- [17] Towards Data Science. *Understanding the concept of simple linear Regression*.
2018. <https://towardsdatascience.com/understanding-the-concept-of-simple-linear-Regression-a572087c253>
Hämtad (2018-06-16)
- [18] Stack Exchange. *Explanation on the independent variable in a logistic Regression*. <https://stats.stackexchange.com/q/250409>
Hämtad(2018-06-16)
- [19] Towards Data Science. *KMeans Clustering for Classification*. 2018.
<https://towardsdatascience.com/kmeans-Clustering-for-classification-74b992405d0a>
Hämtad (2018-06-16)

8 TERMINOLOGI

Amazon Rekognition	Tjänst för bildanalys som Amazon tillhandahåller.
ARFF-format	Attribute Relation File Format. Samling data på specifikt format som behövs vid användning av WEKAs maskininlärningsalgoritmer.
Approximering	Uppskattning av framtida värden.
Attribut	Beskriver en egenskap hos en bild.
Centroid	Mittpunkt i kluster.
Clustering	Metod inom oövervakad maskininläring där datapunkter delas in i kluster.
Kluster	Samling datapunkter.
Klassificering	Datapunkter som delas in i klasser.
Klassifikator	klassificerar objekt genom att placera dem i kategorier.
Regression	Metod inom övervakad maskininläring som används för att approximera värden.
Parameter	Ett värde som tillhör ett attribut.
WEKA	Maskininlärningsbibliotek i JAVA.
Instans	En datapunkt.

