1. ***Hur är AI, Maskininlärning och Deep Learning relaterat?***

AI, Maskininlärning och Deep Learning hör ihop. Utan AI så finns inte maskininlärning och utan AI och Maskininlärning skulle det inte finna någon Deep Learning. För att datorn ska utföra uppgifter som kräver mänsklig intelligens måste den använda olika algoritmer som brukar vara Maskininlärning. Maskininlärning är den delen där algoritmer lär sig från data för att beslut utan en avancerad kod eller backend. Deep Learing är en del av Maskininlärning som använder djupa neuralt nätverk med neuroner för att lösa mönster från data, som exempel: är det en tiger på bilden?

1. ***Hur är Tensorflow och Keras relaterat?***

Tensorflow är en open-source Deep Learning plattform för att bygga och träna neuralt nätverk. Med Keras kan man också träna neuralt nätverk men gör det enklare att skapa och utveckla med olika nätverk exempel: skapa ett bildklassificeringsnätverk där man integreras med Tensorflow och Keras.

1. ***Vad kännetecknar regressionsproblem?***

Regressionsproblem kännetecknar att få ut ett kontinuerligt värde exempel: förutsäga om lokaler är dyrare närmare staden och när liggande transport. Med det skapar modellen en funktion för att skapa en kontinuerlig output.

1. ***Vad kännetecknar klassificieringsproblem?***

Klassificeringsproblem kännetecknar att få ut data tilldelade i olika klasser/kategorier. Exemple: hos en bank, är denna transaktion bedrägeri? “inte bedrägeri” = 0 och “bedrägeri” = 1, med olika omständlig heter så kan modellen lära sig utav ingångsdata och kategoriserar sannolikheten utifrån det.

1. ***Vad är en parameter? Vad är en hyperparameter?***

Parameter är vikter och bias som justerar under upplärningen så man minskar RMSE.

Hyperparameter påverkar upplärningen men är samma efter den har blivit tränat är exemple antal neuroner i ett lagar.

1. ***När man skall göra modellval och modellutvärdering så kan man använda ett tränings, validerings och test data. Förklara hur de olika delarna kan användas.***

Träningsdata tränar modellen genom att justera parameter och vikter. Validering utvecklar hyperparameterna för att få bäst prestation under träningen. Efter träningen användes testdata som utvärdera modellens prestation efter träning.

1. En bild som visar text, Teckensnitt, skärmbild

   Automatiskt genererad beskrivning***Förklara vad kod gör:***

**N\_cols = X\_train.shape[1]**

Med denna kod så tilldelar vi X\_train data matriser till attributet med 1 så tar vi ut index på andra.

**Nn\_model.add()**

Med denna funktion lägger vi till lager i vårat neural nätverksmodell.

**Nn\_model = Sequential()**

Med denna skapar vi ett neuralt nätverk och Sequential är en funktion från Keras, som är en sekventiell modell för neuralt nätverk.

**Nn\_models.add(Dense(100, activation = 'relu', input\_shape =(n\_cols,)))**

Med denna kod så lägger vi in ett dolt lager som har 100 neuroner. Med activation relu så använder vi rectified linear unit aktivation funktion. Input\_shape är tränings data som ska gå igenom funktionerna.

**Nn\_model.add(Dropout(rate=0.2))**

För att kunna använda reglering i Keras så använder man dropout. Med rate 0.2 så inaktiverar slumpmässigt 20% av träning steget.

Efter kommer Dense återigen för att göra hidden layers som är 50 och kör med relu

Sista är en neuronen som blir outputet som används Sigmoid, det är en aktiverings funktion. Med Sigmoid så ändrar man varier 0 och 1 till negative och positiv som är för icke lineära neurala nätverk.

**Nn\_model.compile(optimizer='adam', loss='binary\_crossentropy', metrics=['accuracy'])**

För att kunna kompilera en neural nätverksmodell så bryter man ner den i delar för att utvärdera och optimera modellen.

Optimizer=’adam’ optimeringsalgoritm som är snabb och effektiv mot lösningar inom neuralt nätverk.

Loss= ‘binary\_crossentropy’ är en förlustfunktion som mäter ut den predikterade mellan det faktiska värdet. För att göra optimeringsprocessen mer bättre så användes denna loss

Metrics=['accuracy’] denna så validerar vi olika modeller som är bäst presterande i denna klassificering.

**Early\_stopping\_monitor = EarlyStopping(patience=5)**

För att avsluta träningen om den insisterar kraven så användes early stopping i keras. Det är en parameter som kollar epocher i träningsområden. I detta fall 5 så slutar träningen om den är under 5 epocher än förra. Den är till för att förhindra overfitting.

**Nn\_model.fit(X\_train, y\_train, validation\_split = 0.2, epcohs = 100, callbacks = [early\_stopping\_monitor])**

Sista koden så tränar vi modellen X\_train och y\_train och en split på 20% av träningsdata kommer avända för validering. Epochs är hur många gånger den ska tränas som är 100. Callbacks så skriver den ut funktion early stopping och avslutar koden om den inte hittar någon förbättring.

1. ***Vad är syftet med att att regularisera en modell?***

Ett syfte med regularisering är att förebygga over-fitting så att modellen inte blir beroende under träningsdata som skapar komplikationer under nya data.

1. ***”Dropout” är en regulariseringsteknik, vad är det för något?***

Dropout är som sagt en regulariseringsteknik som förebygger over-fitting under träningen. Med dropout pressar man att inte ta med vissa delar av neuroner som är slumpmässigt utvalda.

1. ***”Early stopping” är en regulariseringsteknik, vad är det för något?***

Early stopping är som sagt en regulariseringsteknik som stoppar träningen om modellens prestation på valideringsdatan blir sämre.

1. ***Din kollega frågar dig vilken typ av neuralt nätverk som är populärt för bildanalys, vad svarar du?***

Den vanligaste och mest använda bildanalys neurala nätverk är Convolutional Neural Networks (CNN). För att den delar upp bilderna i olika layers, enklare och svårare för att identifiera vad som är i bilden. Den hittar mönster i bilden enkla som exempel ögon sen bygger upp det till svårare layers som exempel hela objektet och ger output vad det kan vara på bilden.

1. ***Förklara översiktligt hur ett ”Convolutional Neural Network” fungerar.***

CNN söker efter mönster från bilderna genom att använda konvolutionslayer, hitta dem enkla layers (dem små delarna) och sen bygga upp det till svårare layers (dem större delarna).

1. ***Din vän har ett album med 100 olika bilder som innehåller t.ex. tennisbollar och zebror. Hur hade han/hon kunnat klassificera de bilderna trots att han/hon inte har någon mer data att träna en modell på?***

Jag hade anvisat min vän att använda en pretrained modell och anpassa den till hans behov. Den lär utav pretrained att särskilja olika objektet i albumet exempel är det en tjej eller kille på bilden.

1. ***Din kollega frågar dig vilken typ av neuralt nätverk som är lämpligt för att genomföra en sentimentanalys, vad svarar du?***

Jag hade anvisat min kollega att använda Recurrent Nueral Network för att göra en sentimentanalys, den är bra för sekventiella data som text och ljud.

1. ***Förklara översiktligt hur ett ”Recurrent Neural Network” (RNN) fungerar.***

RNN använder minnen som det har haft innan och gör en prediction utav den minnet sen ger feedback från varje steg om det stämmer.



En bild som visar text, Teckensnitt, skärmbild, linje

Automatiskt genererad beskrivning  
  
  
***Vad gör koden :***

**model.save('model\_file.h5')**

Man sparar tränade modell i en fil ‘model\_file.h5’.

**my\_model = load\_model('model\_file.h5')**

Denna så hänvisar vi och använder tidigare trängde modell som är sparad i .h5 format.

Att använda dessa två underlättar att använda samma träning återigen samt sparatid.