

Stock Market Price Prediction

Tim:

- Matija Aleksić

Zadatak:

Na osnovu podatak cena akcija na berzi firme Tesla za prethodnih 10 godina prikupljenih sa sajta Yahoo Finance (<https://finance.yahoo.com/>), napraviti model koji će da daje sto preciznije predikcije kretanje ovih cena u prethodne dve godine. Od svih podataka iskoristiti prvih 8 godina za treniranje modela, nakon čega će se predikovati cena akcija u poslednje 2 godine.

Analiza podataka:

Nakon učitavanja podataka o kretanju cena akcija potrebno je bilo prostudirati šta znače oznake koje su nam date uz ove podatke.

	Date	Open	High	Low	Close	Adj Close	Volume
0	2012-07-05	6.162000	6.334000	6.160000	6.246000	6.246000	6269000
1	2012-07-06	6.198000	6.346000	6.160000	6.198000	6.198000	3922500
2	2012-07-09	6.188000	6.366000	6.134000	6.298000	6.298000	4552500
3	2012-07-10	6.308000	6.496000	6.178000	6.254000	6.254000	3792000
4	2012-07-11	6.314000	6.336000	6.202000	6.302000	6.302000	3193000
...
2510	2022-06-27	748.099976	756.210022	727.700012	734.760010	734.760010	29726100
2511	2022-06-28	733.450012	749.909973	697.030029	697.989990	697.989990	30130400
2512	2022-06-29	691.500000	693.520020	666.820007	685.469971	685.469971	27632400
2513	2022-06-30	673.530029	688.369995	656.590027	673.419983	673.419983	31533500
2514	2022-07-01	681.000000	690.690002	666.359985	681.789978	681.789978	24781500

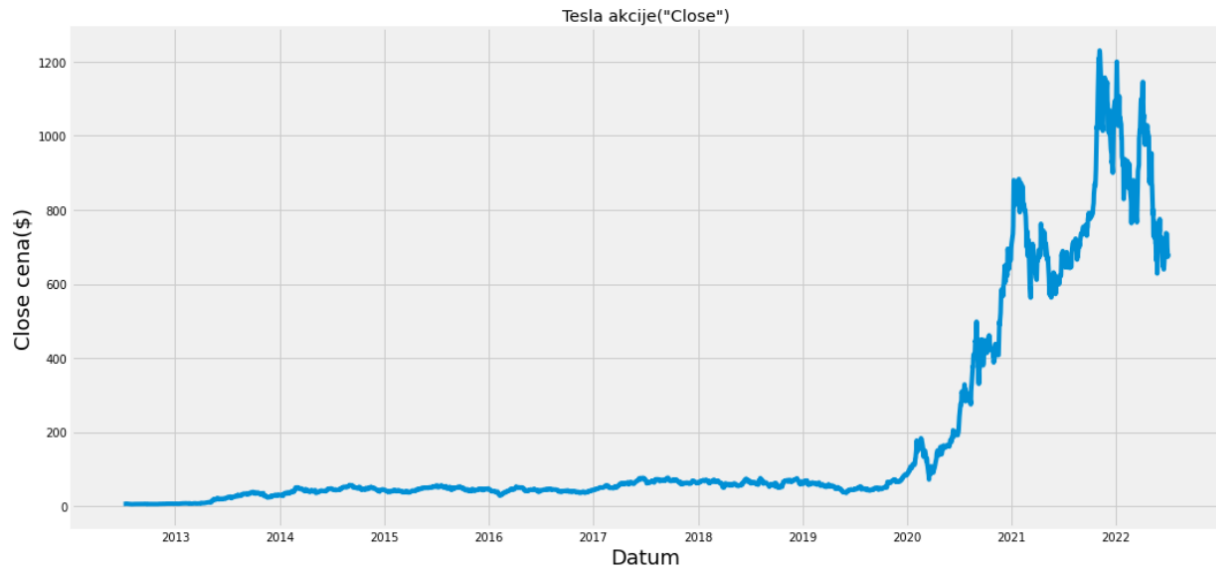
- **High** – maksimalna cena zabeležena u jednom danu
- **Low** - minimalna cena zabeležena u jednom danu
- **Open** - na kojoj ceni je akcija počela na početku dana
- **Close** - na kojoj ceni je akcija završila na kraju dana
- **Volumen** - količina akcija koja je prosla kroz berzni sistem
- **Adj Close** - podešena cena nakon uračunavanja bilo kakvih korporativnih akcija

Nakon proučavanja ovih oznaka odlučili smo da će predikcija našeg modela biti fokusirana na kolonu (**'Close'**) jer ona predstavlja cenu akcija na kraju dana koja su u sustini najbitinija.

Podaci ove kolone će biti skalirani sa **MinMaxScaler**-om u domenu velicina od [0, 1].

Podaci će se takođe podeliti na testni i trening skup podataka. Prvih 8 godina je uzeto da bi se istrenirali modeli ovoga projekta, dok će preostale dve godine biti u testnom skupu odnosno skup koji će biti korišćen za predviđanje podataka.

Grafički prikaz ovih podataka kroz 10 godina izgleda ovako:



Primenjeni modeli:

Modeli koje smo pokušali da implementiramo da bi dobili sto precizniju predikciju su iz početka bili linearni modeli koji nisu davali ni približno rezultatima koji se mogu smatrati zadovoljavajućim. Neke od regresija koje smo isprobali su sledeće:

- **Simple Linear Regression**
- **Lasso Regression**
- **Elastic Net Regression**

Daljim istraživanjem smo zaključili da ovakve vrste predikcija mora raditi putem neuronskih mreža jel su podaci previse nekonzistentni i priodični.

Modeli nerunoskih mreza koje smo isprobali su:

- **Recurent Neural Network(RNN)**
- **Gated recurrent units (GRU)**
- **Long Short-Term Memory(LSTM)**

Izabrani modeli:

Nakom testiranja prethodno navedenih metoda zaključeno je, da je svaka od njih davala zadovoljavajuće rezultate. Optimizator za svaki model koji je bio korscen je **'adam'**, a funkcija gubitka je bila takođe ista za sve modele i ona je **'mean_squared_error'**.

1. RNN Model:

Hiperparametri koji su davali najbolje resenje za model RNN su sledeći:

```
model3.add(SimpleRNN(55, input_shape=(x_train.shape[1], 1), activation="relu", return_sequences=True))
model3.add(Dropout(0.2))
model3.add(SimpleRNN(45, activation="relu", return_sequences=True))
model3.add(SimpleRNN(25, activation="relu"))
model3.add(Dense(10, activation="relu"))
model3.add(Dense(1))

#Kompilacija modela
model3.compile(optimizer='adam', loss='mean_squared_error')
```

Prvi nivo rekurentne nerunske mreže je imao 55 neurona, posle čega sledi **Dropout** layer koji će ignorisati 20% neurona što znači da njihova aktivacija daljih neurona u toku će privremeno biti isključena i takodje podešavanje težina u backward pass-u. Posle toga ide još dve rekurentne mreže sa 45 i 25 neurona, nakon čega ide 10 neurona **Dense** nivoa koji je umrežen sa svim neuronima iz prethodnog nivoa, i na kraju zadnji neuron koji predstavlja predikovanu vrednost.

2. GRU Model:

Hiperparametri koji su davali najbolje resenje za model GRU su sledeći:

```
#Postavljanje i testiranje Model2
#####
model2 = Sequential()
model2.add(GRU(65, return_sequences=True, input_shape=(x_train.shape[1], 1)))
model2.add(Dropout(0.2))
model2.add(GRU(55, return_sequences = True))
model2.add(Dropout(0.2))
model2.add(GRU(25))
model2.add(Dense(1))

#Kompilacija modela
model2.compile(optimizer='adam', loss='mean_squared_error')
```

Prvi nivo GRU neuronske mreže je imao 65 neurona, zatim sledi Dropout nivo od 0.2 za ignorisanje određene količine neurona, zatim opet GRU sloj sa 55 neurona, nakon čeka još jedan Dropout nivo sa takođe vrednošću 0.2, zatim još jedan sloj GRU sa 25 neurona i jedan krajnji neuron da vrati predikovanu vrednost.

3. LSTM Model:

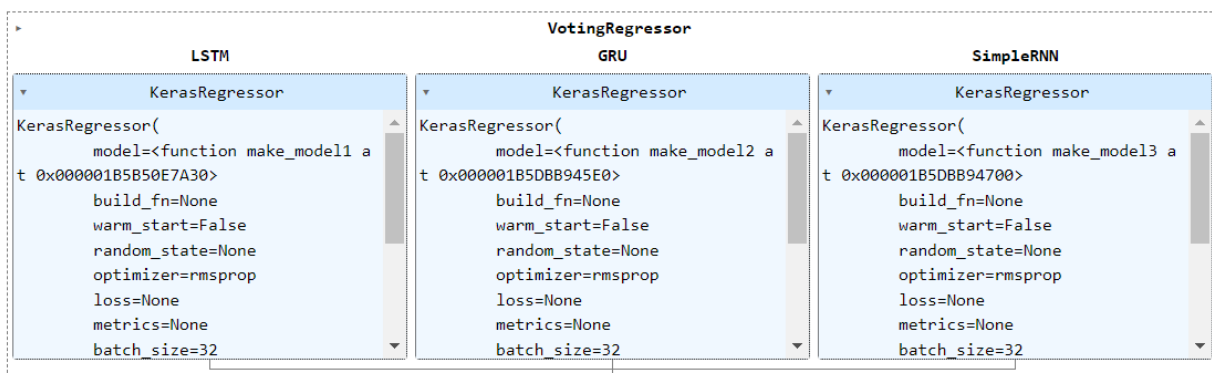
Hiperparametri koji su davali najbolje resenje za model GRU su sledeći:

```
#Postavljanje i testiranje Model1
#####
model1 = Sequential()
model1.add(LSTM(55, return_sequences=True, input_shape=(x_train.shape[1], 1)))
model1.add(Dropout(0.2))
model1.add(LSTM(30, return_sequences = False))
model1.add(Dropout(0.2))
#Dodaje dense layere sa 25 i 1 neuronom
model1.add(Dense(10))
model1.add(Dense(1))

#Kompilacija modela
model1.compile(optimizer='adam', loss='mean_squared_error')
```

LSTM se pokazao kao najbolji model u ovoj trojci i davao je najpreciznije rezultate. Kao i na predhodnim modelim sa slike mozemo da pročitamo koliko je svaki sloj imao neurona a oni su redom 55,30,10,1 uz to da postoje dva Dropout sloja za ignorisanje odredjene proporcije neurona.

Nakon obučavanja svih ovih modela vreme je bilo da ih sve zajedno iskombinujemo da bi dobili još preciznije rešenje. To smo uradili korišćenjem modela Ensambla, konkretno smo koristili model **Voting Regressor-a**, koji je ensamblov meta regresor koji sadrži više baznih modela regresora. U naš voting regresor smo ubacili tri modela navedena u prethodnom delu izveštaja. Ovaj model vadi prosek predikcija ovih modela i kombinuje njihova rešenja kako bi dobilo jedno univerzalno.



Rešenje:

Nakon kombinovanja predikcija sva tri modela dobili smo konačno rešenje koje je zadovoljavajuće i iscrtali smo ga na grafu da bi videli koliko približno je naš model napravio predikcije.

