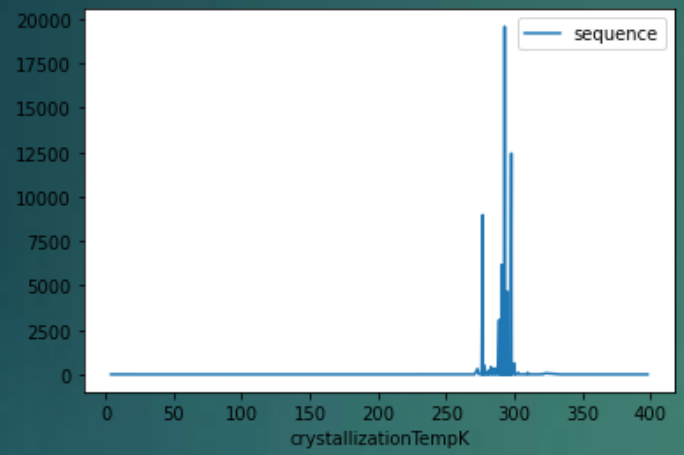
Fehérjék kristályosodási hőmérsékletének megjóslása gépi tanulási eszközökkel

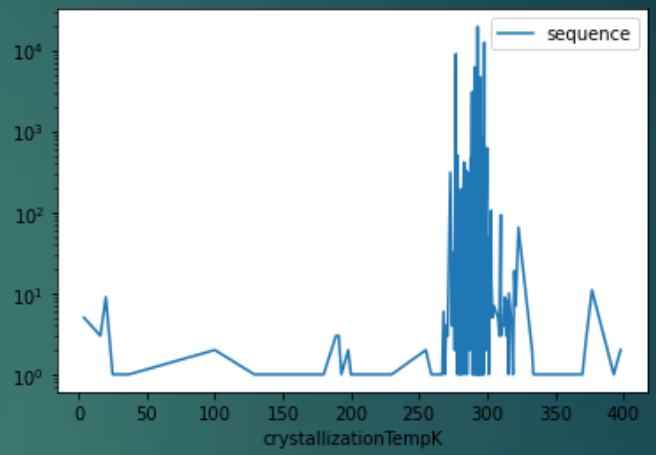
FISCHER KORNÉL

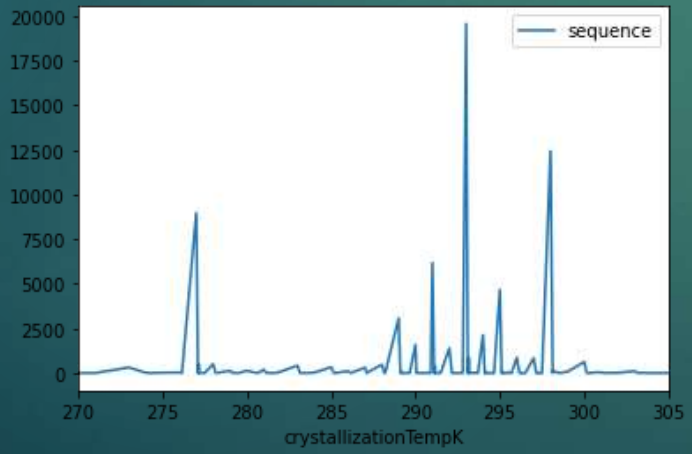
Feladat ismertetése

- Fehérjék kristályosodási hőmérsékletének vizsgálata
- Az input az aminosavak sorrendje, az output a kristályosodási hőmérséklete
- Felhasznált modell a Random Forest és az XGBoost

A hőmérsékletek eloszlása







Adat előkészítése

- Szekvenciák lerövidítése 100 hosszúra:
  - A túl rövid szekvenciákat eldobtam
  - A túl hosszúaknak csak az első száz elemét tartottam meg
  - Ezt a módszert később lehet fejleszteni
- One-hot encoding: minden betűt egy 24 dimenziós vektorral helyettesítünk:
  - 24 féle aminosav fordul elő, bár ezek közül 3 nagyon ritka
- Így egy bemenetet egy 2400 hosszú vektorral reprezentálunk
- Singular Value Decomposition (SVD) használata a dimenziószám 50-re csökkentésére
- Ritka mátrix formátumban tárolható

Singular Value Decomposition

- $A = U * S * V^T$ ahol:
  - A az eredeti mátrix, mérete $n * m$
  - S főátlójának elemei a $\sqrt{\lambda_i(A * A^T)}$-k, ahol $\lambda_i(B)$ a B mátrix i. legnagyobb sajátértéke. S mérete $n * m$
  - U oszlopai az A mátrix bal szinguláris vektorai, V oszlopai pedig a jobb szinguláris vektorok. U $n * n$, V pedig $m * m$ méretű
- Gyakran alkalmazzák, ha egy mátrix dimenziószámát szeretnék csökkenteni, én is erre használtam
- Az U első 50 oszlopát tartottam csak meg

Random Forest

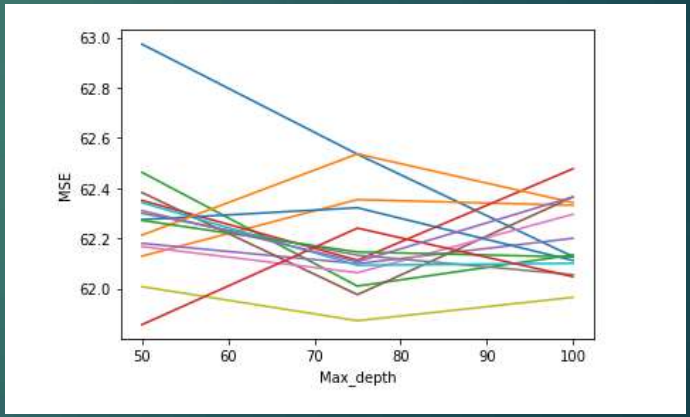
	max_depth	min_samples_split	min_samples_leaf	mse
0	50	25	3	64.541878
1	50	25	4	64.376490
2	50	25	5	63.933511
3	50	50	3	63.880515
4	50	50	4	63.814165
5	50	50	5	63.380421
6	50	75	3	62.910865
7	50	75	4	62.905504
8	50	75	5	62.857215
9	75	25	3	64.793728
10	75	25	4	64.433140
11	75	25	5	63.295134
12	75	50	3	64.245540
13	75	50	4	63.522753

Hiperparaméterek állítása

```
#A Random Forest kiértékelése
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error

Y = joined_list['crystallizationTempK']
X = onehot_encoded_sparse_transf
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.1, random_state=42)

model = RandomForestRegressor(criterion = 'mse', max_depth = 100, min_samples_split = 25, min_samples_leaf= 10)
reg=model.fit(X_train, Y_train)
pred= reg.predict(X_test)
mse = mean_squared_error(Y_test, pred)

print("The mean squared error (MSE) on test set: {:.4f}".format(mse))
```

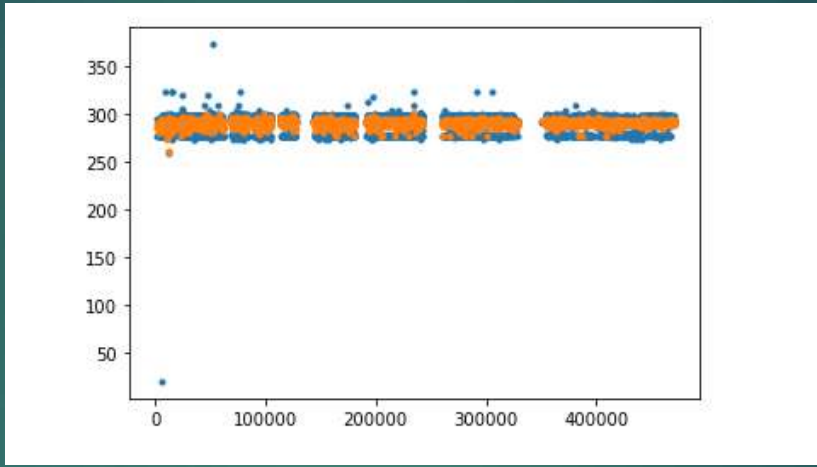
```
The mean squared error (MSE) on test set: 57.1986
```

A modell kiértékelése a legjobb paraméterek mellett



Az elért legkisebb mse: 57.1986.

Random Forest



A hőmérsékletek átlaga 291 fok volt, a medián és módusz 293. Kipróbáltam, milyen mse érték adódna, ha a jóslat végig ezen két érték egyike lenne. Ezekben az esetekben az mse 63 illetve 67 volt. Vagyis a Random Forest javított ehhez képest.

XGBoost

```
#Kiértékelés
from sklearn.model_selection import train_test_split

Y = joined_list['crystallizationTempK']
X = onehot_encoded_sparse_transf
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.1, random_state=42)

xg_reg = xgb.XGBRegressor(objective ='reg:linear', colsample_bytree = 0.3, learning_rate = 1, max_depth = 5, alpha = 10, n_estimators = 10)

xg_reg.fit(X_train,Y_train)

preds = xg_reg.predict(X_test)

mse = mean_squared_error(Y_test, preds)
print("MSE: %f" % (mse))
```

```
[16:36:44] WARNING: ../src/objective/regression_obj.cu:174: reg:linear is now deprecated in favor of reg:squarederror.
[16:36:45] WARNING: ../src/objective/regression_obj.cu:174: reg:linear is now deprecated in favor of reg:squarederror.
MSE: 63.301344
```

Az elért legjobb mse: 63.301344.

Az XGBoost esetén nehezebb a hiperparaméterek hangolása, mint a Random Forestnél. Emiatt lehet jobb az eredmény utóbbinál.

Jövőbeli feladatok

- A mostani módszerek javítása:
  - Padelés használata a túl rövid szekvenciáknál, vagy loopolás
  - A ritka aminosavakat ugyanazzal a betűvel jelöljük
- El kellene érni, hogy akármilyen hosszú aminosav szekvenciát be tudjunk adni a rekurzív modellnek, és az jól tudjon jósolni
- LSTM (Long Short Term Memory) kipróbálása
- BERT modell betanítása a fehérje adatbázisra

Köszönöm a figyelmet!