PROJECT TOPIC

- **Goal:** Create a ELO Converter, converting Lichess Elo Ratings to Chesscom Elo Ratings. Chess has a Strength System called ELO, the higher it is the stronger you are. However, the Elo Systems are contained, so Elo on one website doesn't Translate 1:1 to the other. My goal here is to build a ML Model to translate the ELO from one site to the other.
- **Data**: I have a Dataset of more than 50k ELO Ratings from both website from people with the same username. This needs to be cleaned up as there are many false positives. I gathered this myself for another project and am looking to publish it on Kaggle or similar, however the full set has a lot of personal info, that I will need to check if that needs to be removed.
- **Machine Learning**: I will try K nearest neighbours and Linear Regression and see which works best for this problem. I will attempt different version of the KNN, to find the best parameters for me.

This project is inspired by some Data I gathered a while ago. I gathered the Dataset and wanted to do some ML on it, but never got around to it. So this is a perfect opportunity

The Dataset

The Dataset are ELO Values for about 50k users on lichess and chesscom. I will create a predictor that predicts the rating for one rating to another.

Loading the Data in

	li_bullet_rating	li_bullet_rd	li_blitz_rating	li_blitz_rd	li_rapid_rating	li_rapid_rd	ch_bullet_ra
username							
garabomboelinvisible	1096	106	1043	52	1079	199	
antonym007	1196	45	1523	67	1587	102	
rilikva	1500	500	1411	342	2019	45	
pushydiscovery	2470	47	2424	96	2392	80	2
weaponizedspaghetti	1212	230	884	99	1153	53	

Data Description

```
<class 'pandas.core.frame.DataFrame'>
Index: 118133 entries, garabomboelinvisible to abo01
Data columns (total 13 columns):
 #   Column            Non-Null Count   Dtype
---  ------            --------------   -----
 0   li_bullet_rating  118133 non-null  int64
```

```
 1   li_bullet_rd      118133 non-null  int64
 2   li_blitz_rating   118133 non-null  int64
 3   li_blitz_rd       118133 non-null  int64
 4   li_rapid_rating   118133 non-null  int64
 5   li_rapid_rd       118133 non-null  int64
 6   ch_bullet_rating  59032 non-null   float64
 7   ch_bullet_rd      59032 non-null   float64
 8   ch_blitz_rating   78020 non-null   float64
 9   ch_blitz_rd       78020 non-null   float64
 10  ch_rapid_rating   77151 non-null   float64
 11  ch_rapid_rd       77151 non-null   float64
 12  ch_fide_rating    69432 non-null   float64
dtypes: float64(7), int64(6)
memory usage: 12.6+ MB
```

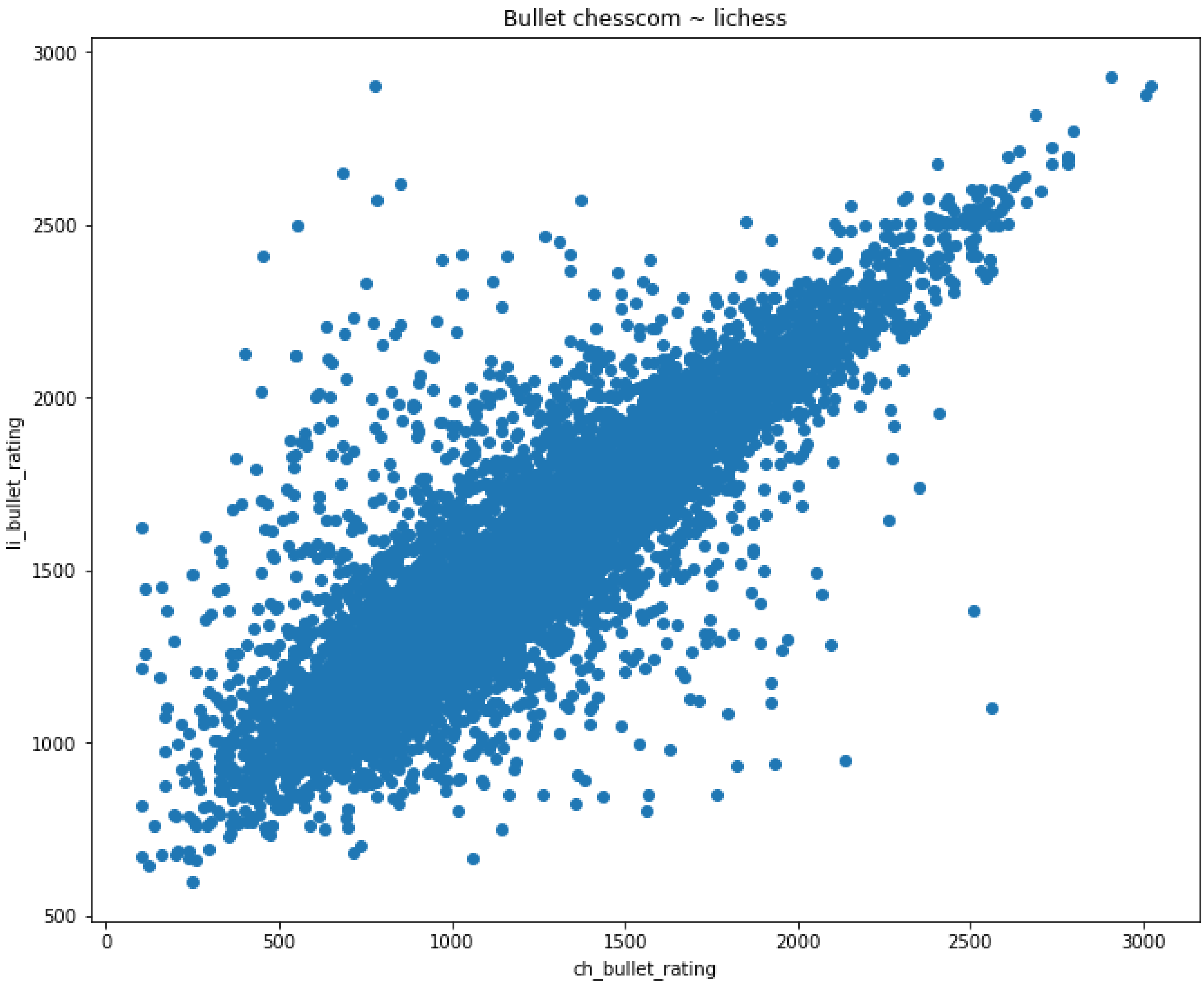
- **RD is the *rating_deviation***
- **li prefix means lichess**
- **ch prefix means chesscom**

- **Bullet, Blitz and Rapid are game modes in Chess**

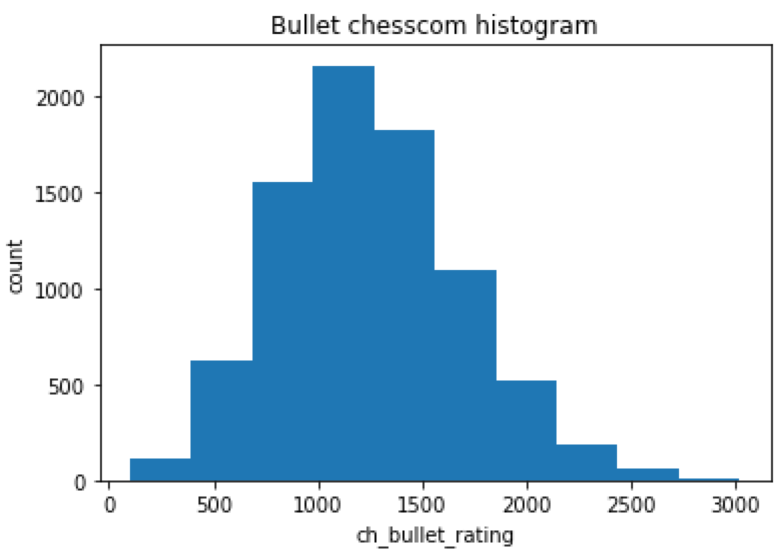
df now contains the interesting columns

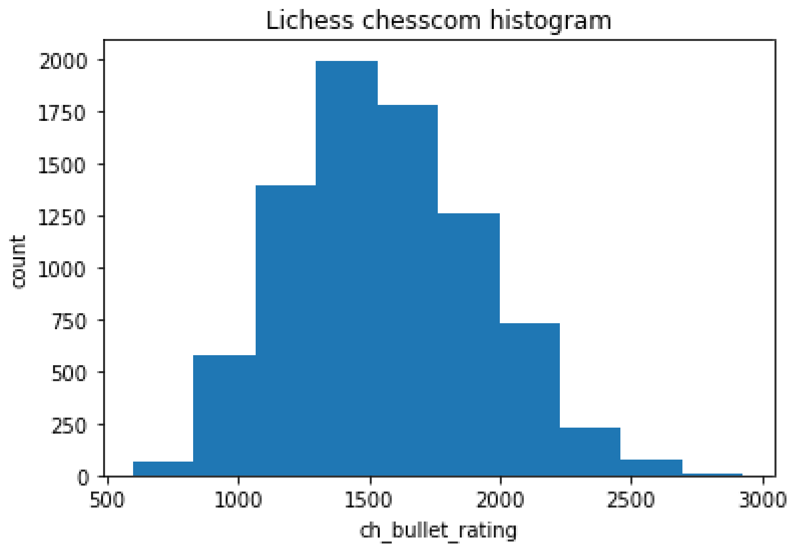
Cleaning the Dataset

Now filter for an RD below 100 to remove untelling values.



there are some outliers but I will clean some up





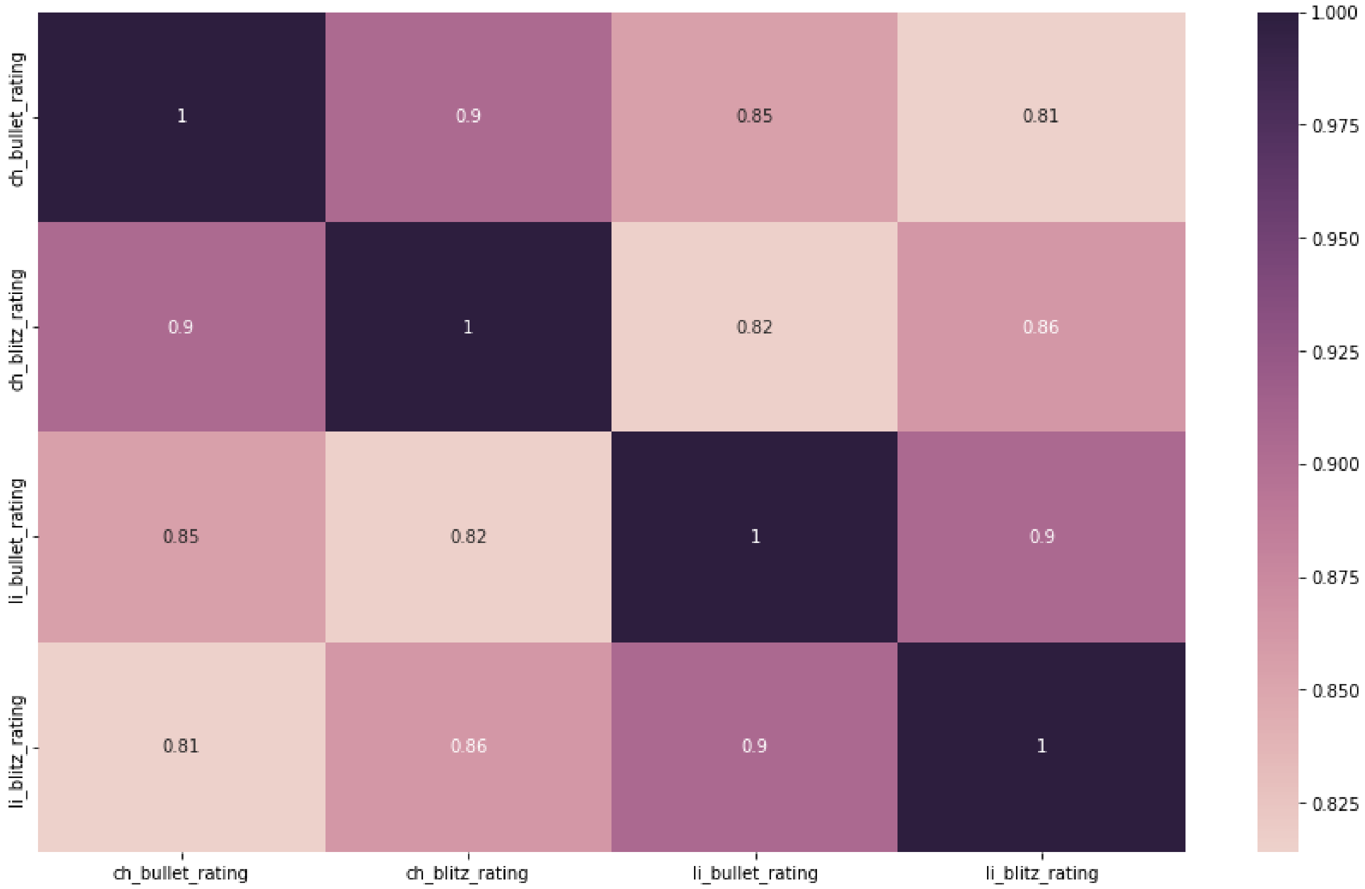
There is a heavy bias towards values between 1000-2000

Looking at Correlations

FIDE, Bullet, Blitz, Rapid are the columns of interest

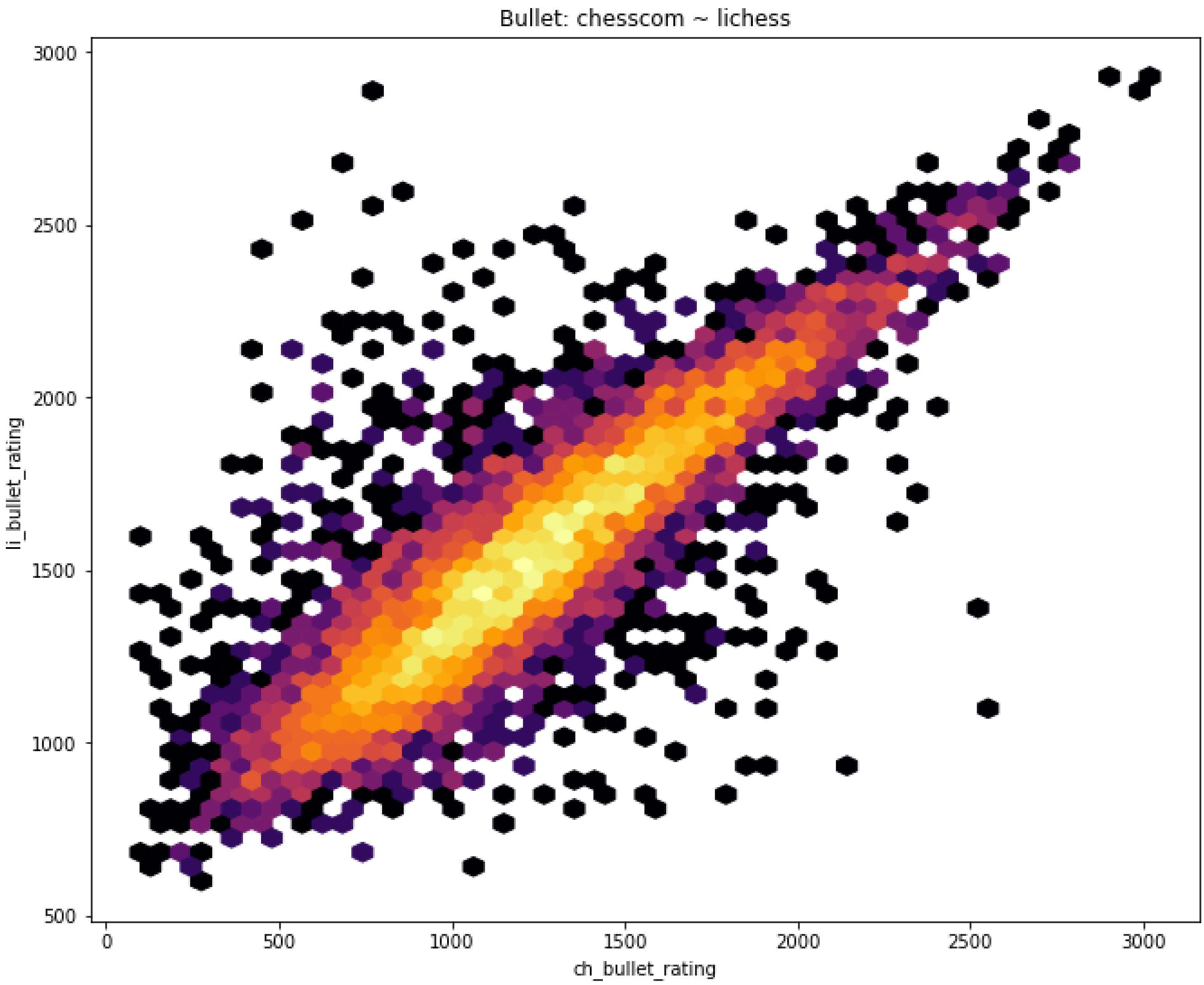
FIDE is our target value

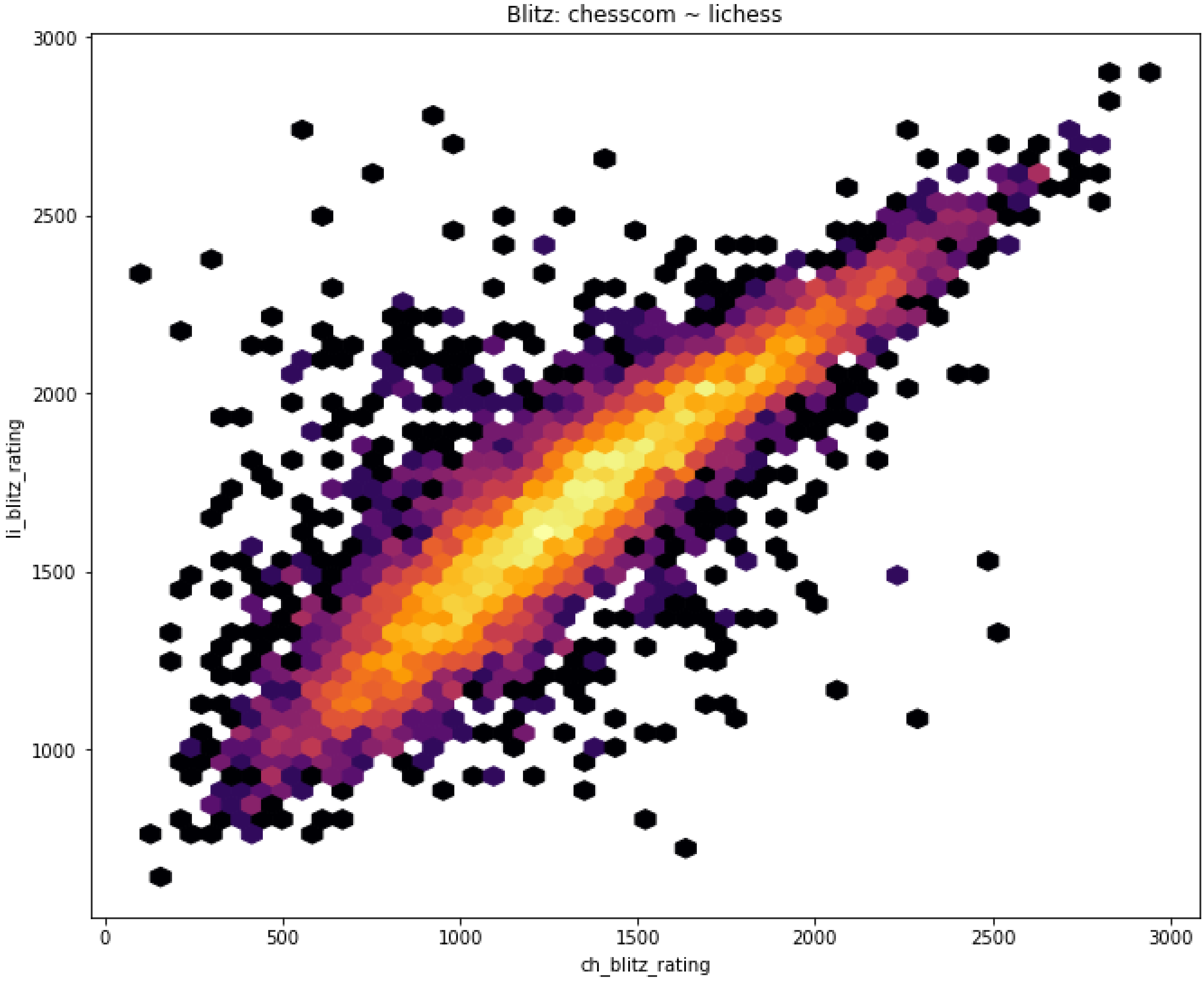
```
<AxesSubplot:>
```



There is decent correlation here. I can see that chess_bullet <-> chess_blitz, lichess_bullet <-> lichess_blitz have the strongest correlations. But all have good relations to eachother.

Visualize Correlations


Bullet: chesscom ~ lichess
li_bullet_rating
ch_bullet_rating
3000
2500
2000
1500
1000
500
0
500
1000
1500
2000
2500
3000



Model Training

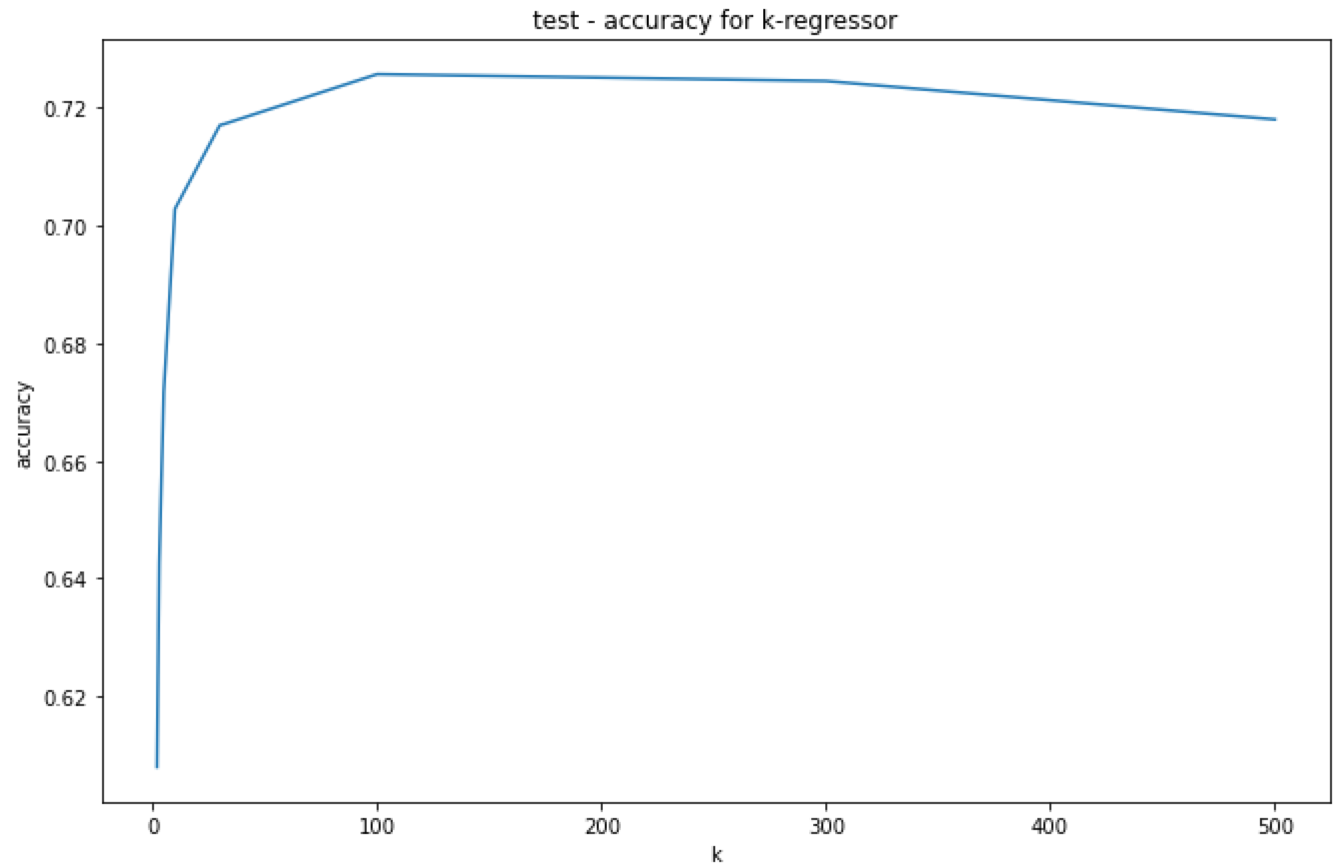
train, test, validation 70,20,10 split

```
length of
 Dataset: 8123
 train_set: 5686
 test_set: 1624
 val_set: 813
```

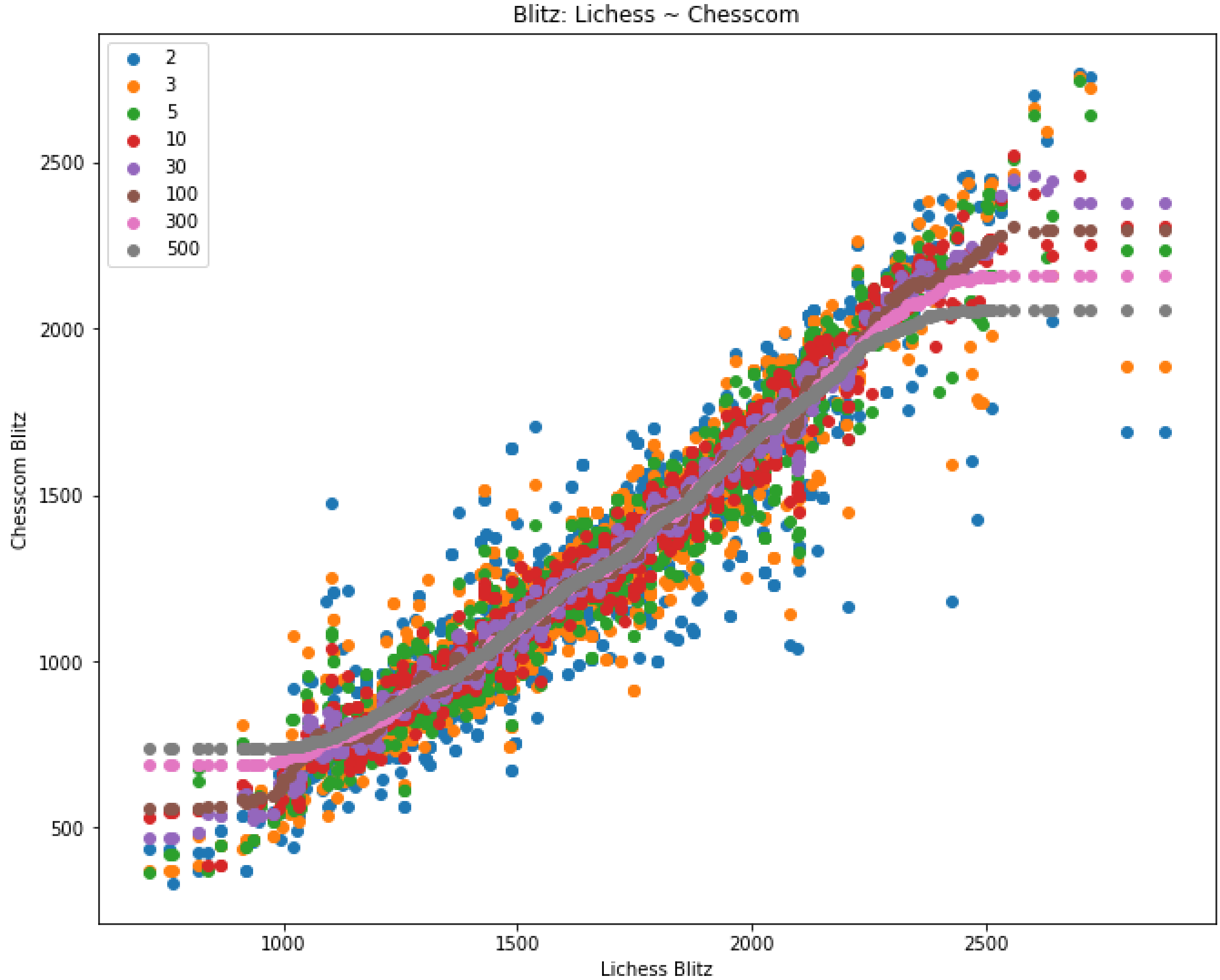
K-nearest-neighbours

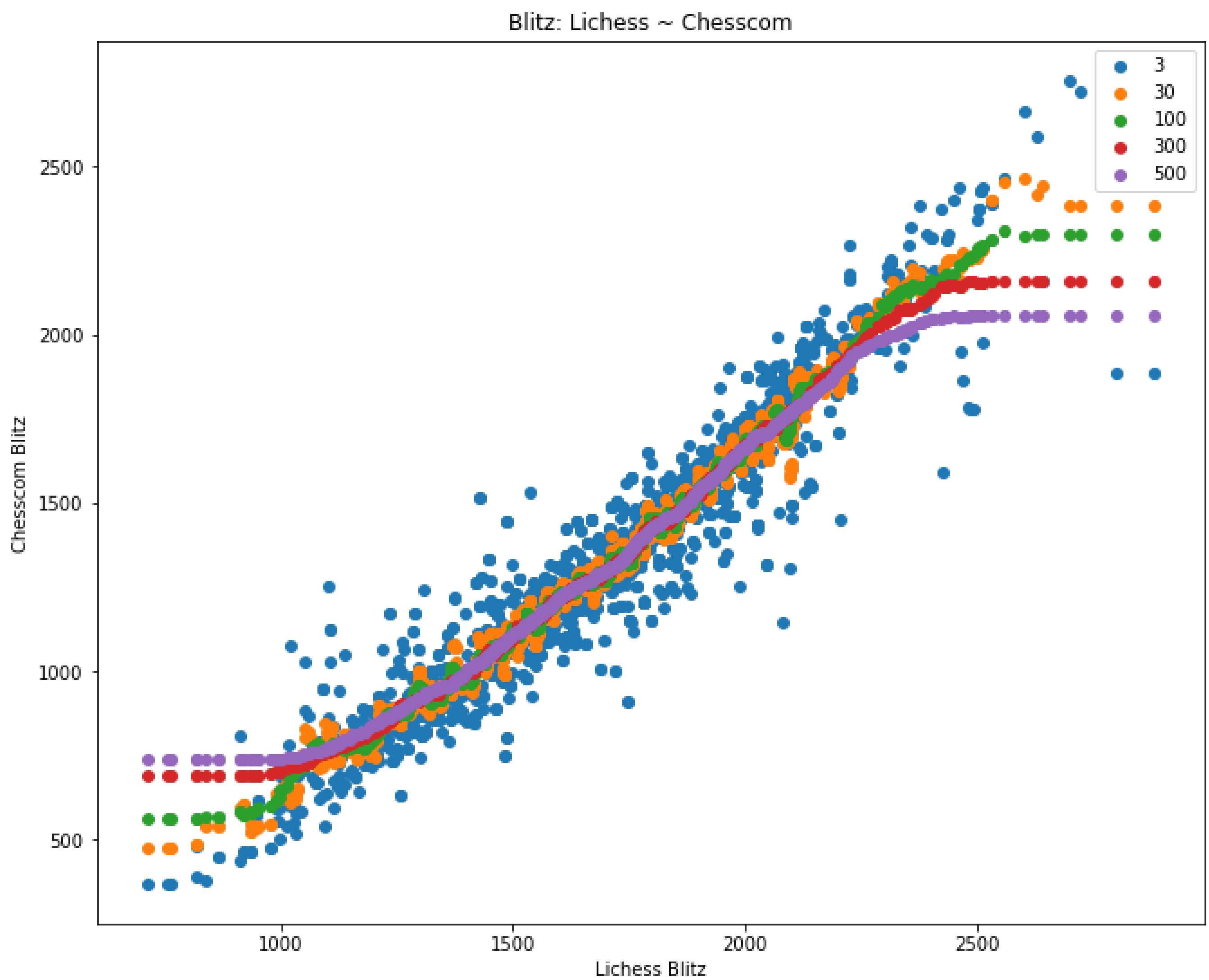
K=2,5,10,30,100,300,500

```
k2 0.6079834524076553
k3 0.6425789977252938
k5 0.6716768201397763
k10 0.7027723435737401
k30 0.7169990987858982

k100 0.7257000668551719
k300 0.724533049090944
k500 0.7180013561487815
```



K=100 seems to be the best


Blitz: Lichess ~ Chesscom
2
3
5
10
30
100
300
500
Chesscom Blitz
Lichess Blitz
500
1000
1500
2000
2500



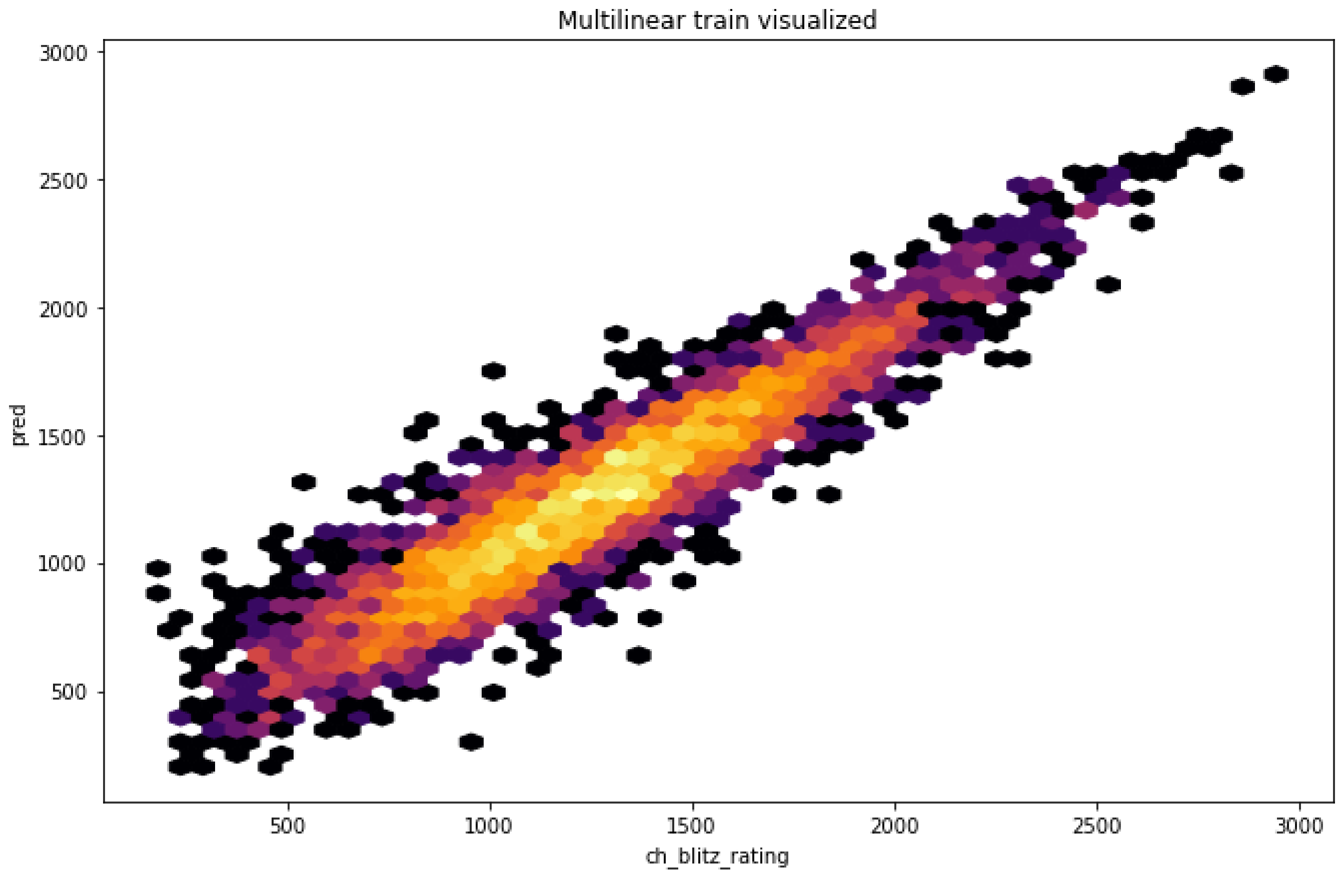
Due to the lack of data for the high values there is some overfitting going on for high k

Linear- and Multilinear Regression

Multilinear for target: FIDE on Blitz and Bullet

```
                            OLS Regression Results
==============================================================================
Dep. Variable:        ch_blitz_rating   R-squared:                       0.871
Model:                            OLS   Adj. R-squared:                  0.871
Method:                 Least Squares   F-statistic:                 1.085e+04
Date:                Sun, 11 Sep 2022   Prob (F-statistic):               0.00
Time:                        21:25:06   Log-Likelihood:                -31200.
No. Observations:                4841   AIC:                         6.241e+04
Df Residuals:                    4837   BIC:                         6.243e+04
Df Model:                           3
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
Intercept         -185.1669     12.117    -15.281      0.000    -208.922    -161.411
li_blitz_rating      0.6730      0.015     44.027      0.000       0.643       0.703
li_bullet_rating    -0.3016      0.016    -18.679      0.000      -0.333      -0.270
ch_bullet_rating     0.6725      0.010     69.114      0.000       0.653       0.692
==============================================================================
Omnibus:                      270.163   Durbin-Watson:                   1.993
```

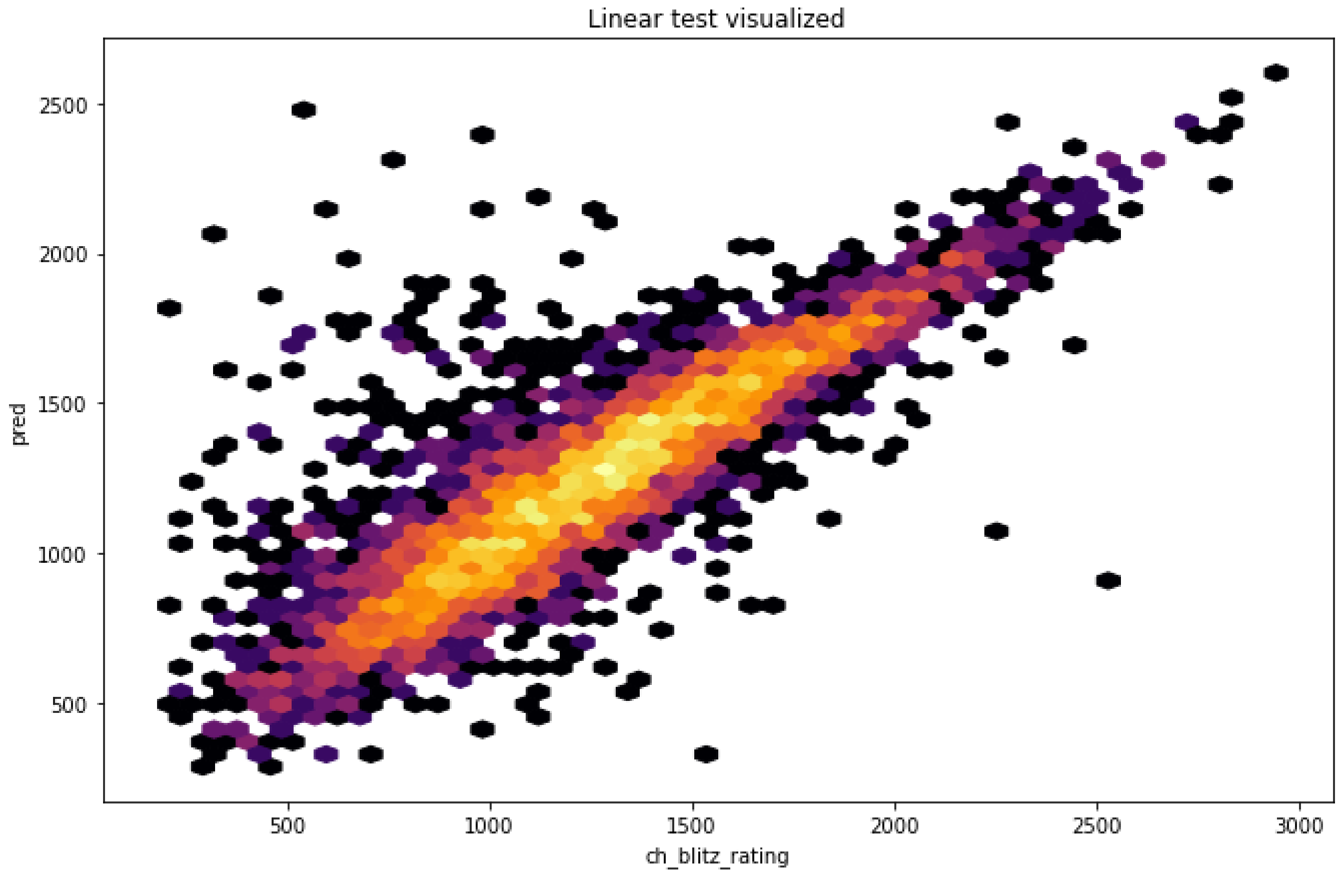
```
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              671.843
Skew:                          -0.324   Prob(JB):                    1.29e-146
Kurtosis:                       4.706   Cond. No.                     1.44e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.44e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```



Linear for target: FIDE on Blitz

```
                            OLS Regression Results
==============================================================================
Dep. Variable:        ch_blitz_rating   R-squared:                       0.736
Model:                            OLS   Adj. R-squared:                  0.736
Method:                 Least Squares   F-statistic:                 1.350e+04
Date:                Sun, 11 Sep 2022   Prob (F-statistic):               0.00
Time:                        21:25:07   Log-Likelihood:                -32925.
No. Observations:                4841   AIC:                         6.585e+04
Df Residuals:                    4839   BIC:                         6.587e+04
Df Model:                           1
Covariance Type:            nonrobust
===================================================================================
                      coef    std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------------
Intercept        -536.9158     15.774    -34.038      0.000    -567.840    -505.992
li_blitz_rating     1.0896      0.009    116.179      0.000       1.071       1.108
==============================================================================
Omnibus:                     1892.213   Durbin-Watson:                   1.973
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            20615.396
Skew:                          -1.554   Prob(JB):                         0.00
Kurtosis:                      12.620   Cond. No.                     8.48e+03
==============================================================================
```

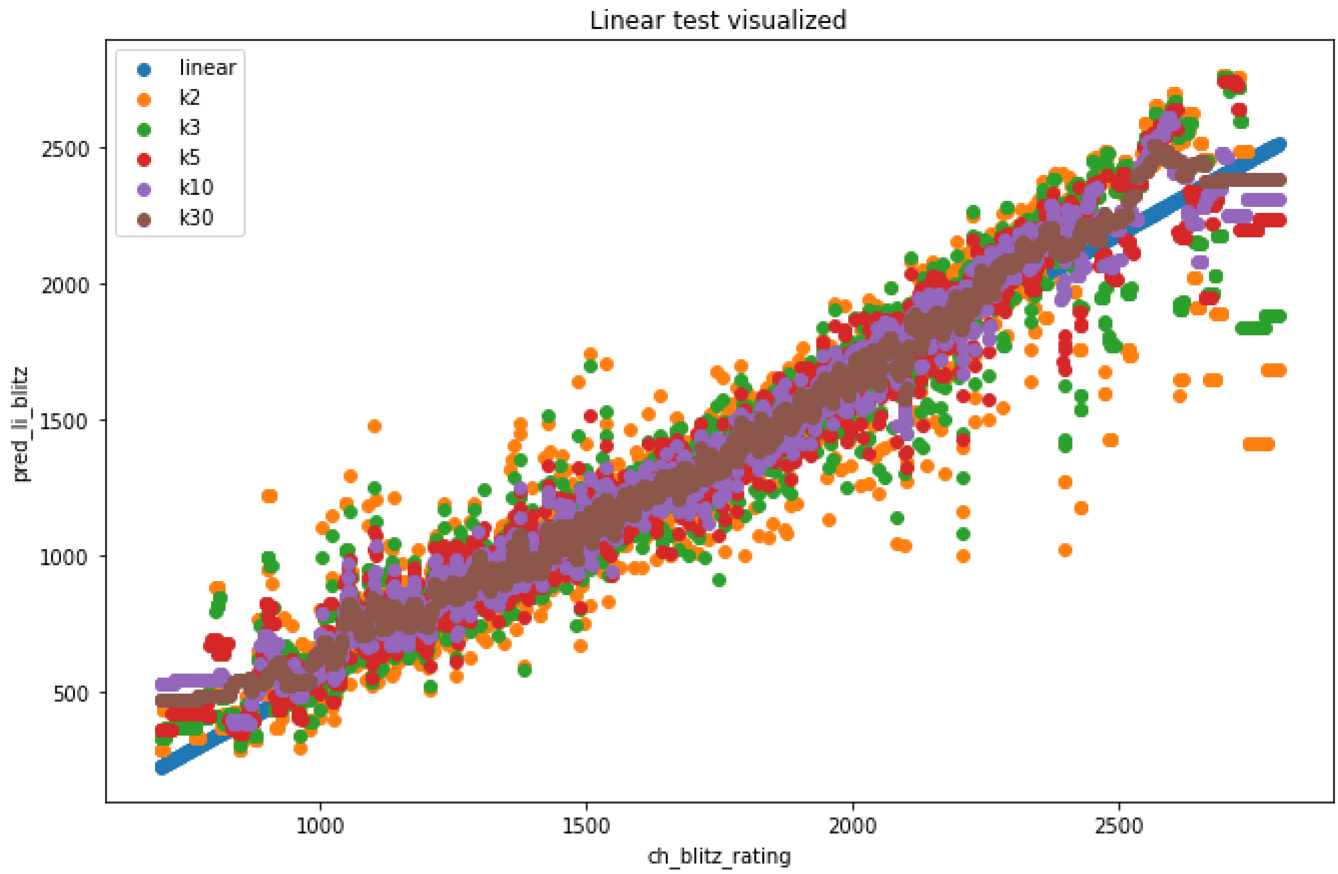
```
Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 8.48e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

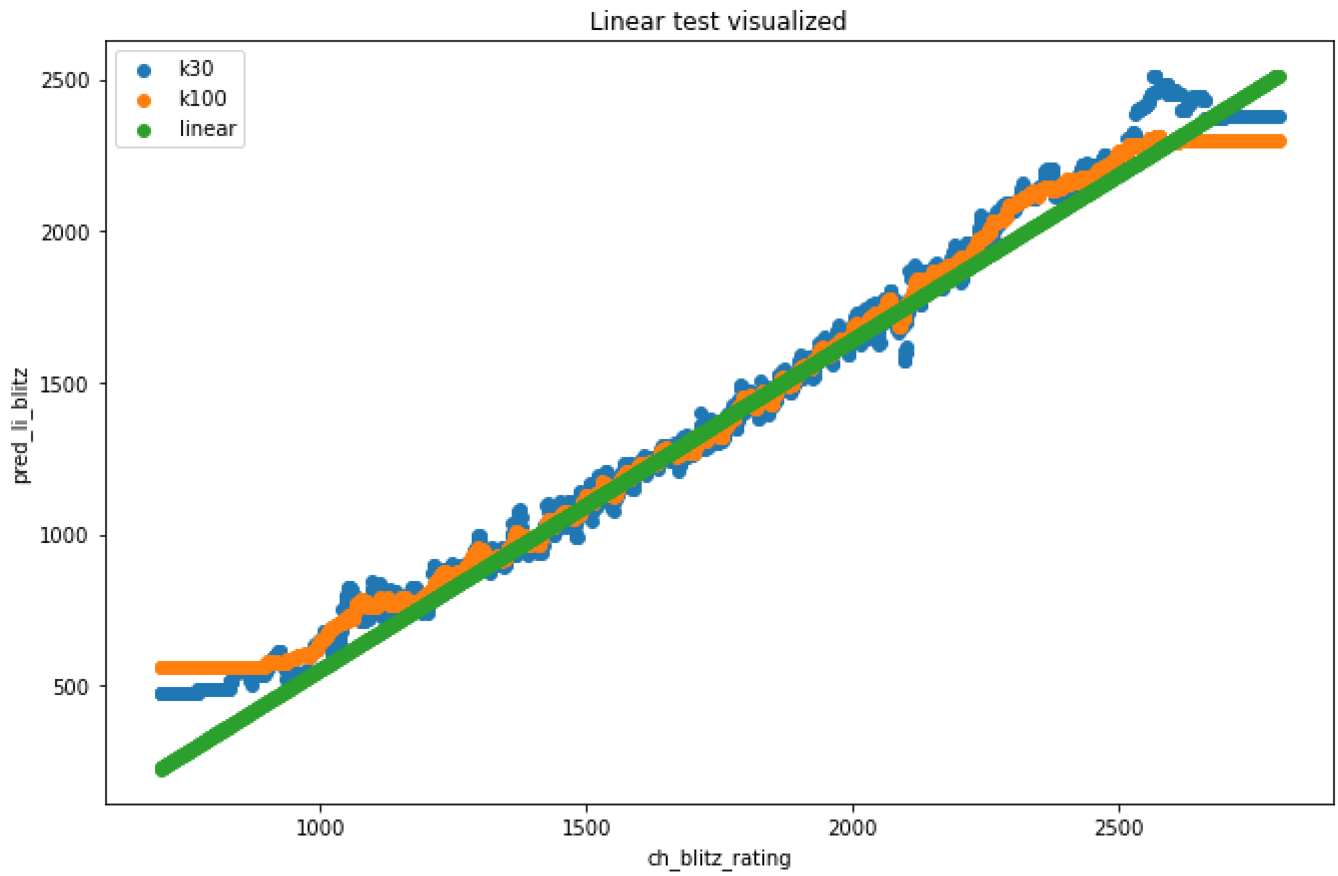


Accuracies

```
r_squared_accuracy of
 MultiLinear: 0.7750097278913504
 Linear: 0.6819650681539863
```

Export the data






```
r-aquared accuracies
k30 0.7169990987858982
k100 0.7257000668551719
k300 0.724533049090944
k500 0.7180013561487815
Linear 0.6819650681539863
```

~75 r-squared accuracy is decent for the Dataset at hand. K100 is the one I am most happy with.

Conclusion

K-nearest-neighbours with k=100 is my Model of choice. It gives the best r-squared accuracy. The only issues is has it that for very low and very high Elos we have don't have very good accuracy. However, most people lie within 700 and 2500 Elo, for which is it extremely accurate.

k=30 is almost as good and can also be used, it is a little better near the edges and almost as good as k=100 for the rest, but a little bit too much variance, leading to illogical results.

Multilinear regression is the best Model, however it depends on more than just one input variable, so if we have more information from the Users, except just one Elo. Then it is the most powerful, however the information gain does not move the needle a lot. We get 5% more accuracy, but require 300% more data.