Using feature extraction, I created 10 new features that might be useful for improving the performance of a behavior detector for "OffTask".

For feature engineering brainstorming, I applied PCA to extract the principal components. The principal components emphasize the variations and patterns in the initial structure of the new data set('ca2-dataset.csv'). From there, I understood the existing features and got an idea of correlation between variables for creating new features. Also, I applied Prof. Baker's suggestions_[some useful types of new features]. https://www.youtube.com/watch?v=hE0fatxFyh4 for brainstorming.

-List of New Features-

1. History of knowledge estimate change: 'Pchange_so_far' & 'right_so_far'
The new features mean cumulative sums of knowledge estimate and correct answers for individual students. The counts might have a positive association with the behavior ontask. I selected the feature 'right', not 'notright' as 'notright' included values on wrong answer and help too, which might indicate both on/offtask.

2. Capturing learning moment: 'pknow-1_right' & 'Pknow-2_right'
'pknow-1' and 'Pknow-2' are negatively correlated with offtask behavior. Combining these features and 'right' can be informative as these new features can capture the fact that the learner knew the skill/knowledge.

3. Percentages of wrong action type: 'pct_wrong'
To find out the proportion of only wrongs to an overall number of wrongs(including 'help' action), I combined 'wrongpct-up' which excluded help as wrong, and 'howmanywrong-up'. I think the ratio of wrong actions is significant. If the ratio is high, it may mean that the learner are struggling with problems or that she/he is just not paying attention. Both suggestions can be highly related to students' outcomes.

4. Time taken for wrong action: tsd_wrong'
The new feature was generated by multiplicative interaction. Longer time taken for last actions and the total number of actions where this production was wrong might indicate a student's learning difficulty which could negatively affect learning progress, potentially leading to offtask behavior.

5. Total counts per last-n-actions: 'conuts-prev3' and 'counts-prev5'
How many involved the same interface widget, especially with last-n-actions might indicate a lot of information on the learner's persistence and progress. Final handling(or recent action) in the problems might positively correlate with a student's ontask. Calculating the total counts of the values overall can make the model more simple.

6. Time taken for last-n-actions: 'tsd_prev3' & 'tsd_prev5'
Longer time taken for 'how many involved the same interface widget' might indicate that a student keeps trying on recent problems, positively correlating with learning persistence and engagement.

I built the new dataset('new_data.csv'), adding the 10 new features on the old dataset('ca1-dataset.csv'). and eventually obtained Mean Cohen's Kappa: 0.95, F1-Score: 0.96, and ROC AUC: 0.75 while modifying and improving the models and adding some new features. However, I'm not sure these measures really reflect how well the new features improve the model because they are comparatively greater than the performance of the original dataset.

```
Mean Cohen's Kappa: 0.31
F1-Score: 0.32
ROC AUC: 0.75
```

Thus, I have to figure out more several metrics to evaluate the model's performance. Regarding these model performances, please share your advice~:)