Dataset Findings:

Source of Dataset: Speed Video Global Operating Platform at Huawei

Nature of Dataset: 89,266 samples

12 features

1 target

Description of Dataset:

1. Average rate of playing phase (kbps): average number of kilobits that users device downloads per second while playing video.
2. Video total download rate(kbps): total amount of data downloaded per second on device.
3. Video Bitrate (kbps): amount of data transferred per second by video
4. Initial max download rate(kbps): maximum download speed allowed for video stream.
5. End to End round trip time (ms): time taken for a packet to travel from the source to destination and back again.
6. Initial Buffer Latency (ms): delay in initial start of playback
7. Video initial buffer downloaded (byte): initial downloaded data of video
8. Playing time(ms): amount of time it takes for video to play
9. Playing total duration (ms): refers to total duration of playback
10. Stalling times: number of times video stalled
11. Stalling duration(ms): duration of stall in video
12. Stalling ratio: Stalling duration/Playing total duration

Dataset Analysis And Pre-Processing

Finding correlations among features and between feature and target.

1. Correlation Analysis:

Using Variance as a basis for finding correlation got following observations:

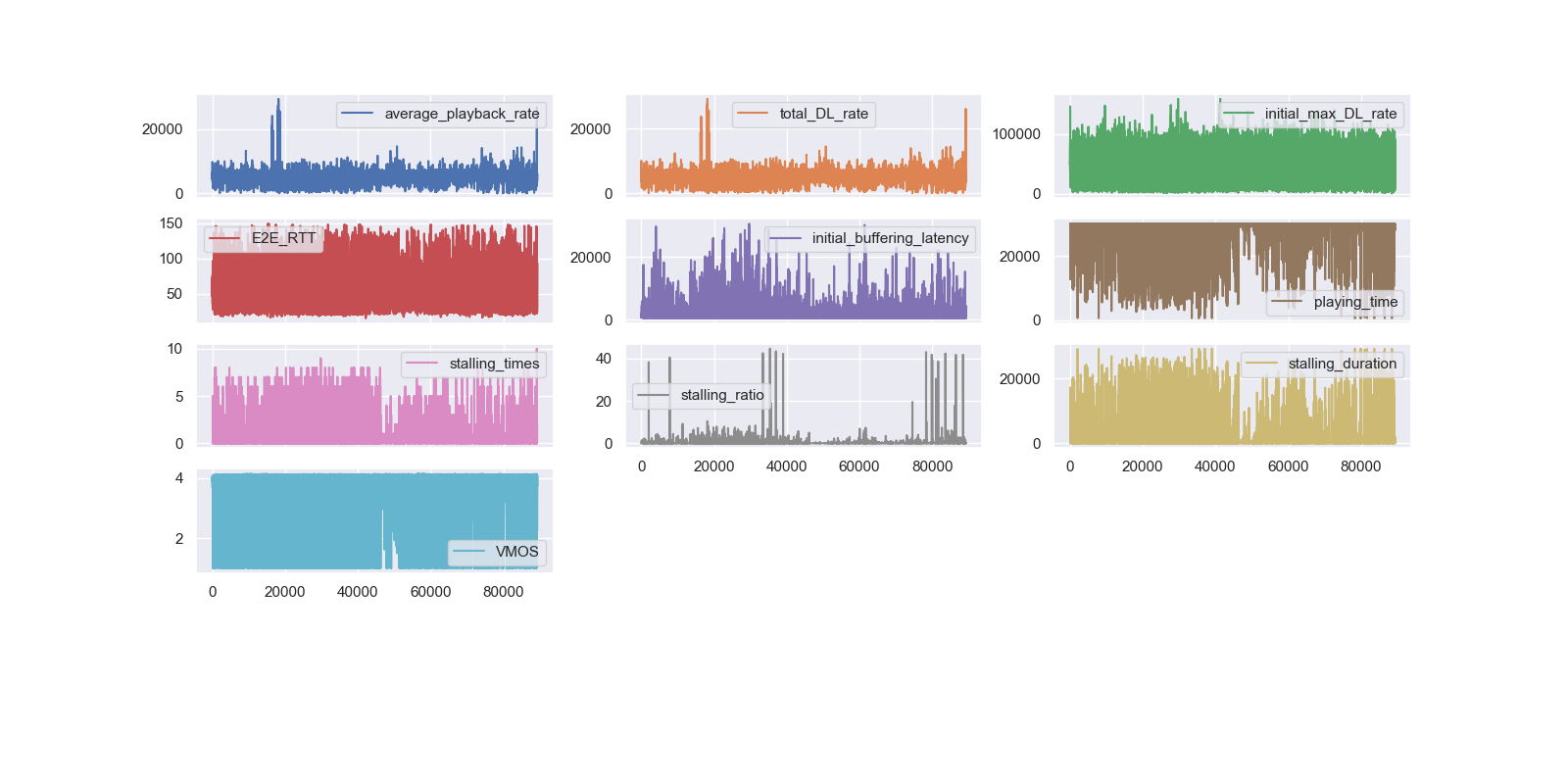
Average playback rate <-> total download rate [Highly Correlated]

Playing time <-> stalling time [Highly Correlated]

Stalling time <-> stalling duration[Highly Correlated]

Video bitrate, End to End round trip time, Playing total duration, Stalling ratio shows very poor correlation with the target.

1. Detection of Outliers:

* Visual method: with the matplotlib library plotted each data on separate plots and got this result.

From the above plots, the spikes in the plot clearly shows presence of outliers in the data.

* Using Inter Quartile Range method:

Taking multiplier = 2, I got

Feature 'average\_playback\_rate' has 6845 outliers = 7.668093114959783 %

Feature 'total\_DL\_rate' has 5796 outliers = 6.492953644164631 %

Feature 'initial\_max\_DL\_rate' has 44 outliers = 0.04929088342706069 %

Feature 'E2E\_RTT' has 2000 outliers = 2.2404947012300314 %

Feature 'initial\_buffering\_latency' has 4183 outliers = 4.6859946676226105 %

Feature 'playing\_time' has 10230 outliers = 11.46013039679161 %

Feature 'stalling\_times' has 4170 outliers = 4.671431452064616 %

Feature 'stalling\_ratio' has 4070 outliers = 4.559406717003115 %

Feature 'stalling\_duration' has 4158 outliers = 4.657988483857236 %

Feature 'VMOS' has 5071 outliers = 5.680774314968745 %

* Using Z-score method:

Taking -3 to 3 as range for Z-score, I got:

Feature 'average\_playback\_rate' has 2531 outliers = 2.835346044406605 %

Feature 'total\_DL\_rate' has 2499 outliers = 2.7994981291869245 %

Feature 'initial\_max\_DL\_rate' has 167 outliers = 0.18708130755270763 %

Feature 'E2E\_RTT' has 1753 outliers = 1.9637936056281229 %

Feature 'initial\_buffering\_latency' has 1304 outliers = 1.4608025452019806 %

Feature 'playing\_time' has 1844 outliers = 2.065736114534089 %

Feature 'stalling\_times' has 2182 outliers = 2.4443797190419647 %

Feature 'stalling\_ratio' has 432 outliers = 0.48394685546568683 %

Feature 'stalling\_duration' has 1844 outliers = 2.065736114534089 %

Feature 'VMOS' has 2795 outliers = 3.131091344968969 %

1. Handling of outliers:

* Using Inter Quartile Range method:

It is observed if outliers removed using this method the correlations obtained has poor strength of the correlation between feature and target.

Proof:

Original Correlation:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Feature Name | average\_playback\_rate | total\_DL\_rate | initial\_max\_DL\_rate | 'E2E\_RTT' | initial\_buffering\_latency | playing\_time | stalling\_times | stalling\_ratio | stalling\_duration |
| VMOS | 0.62 | 0.69 | 0.44 | -0.34 | -0.76 | 0.83 | -0.87 | -0.34 | -0.83 |

Correlation after removing outliers:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Feature Name | average\_playback\_rate | total\_DL\_rate | initial\_max\_DL\_rate | 'E2E\_RTT' | initial\_buffering\_latency | playing\_time | stalling\_times | stalling\_ratio | stalling\_duration |
| VMOS | 0.33 | 0.45 | 0.45 | 0.39 | -0.8 | 0.77 | -0.78 | -0.47 | -0.77 |

Taking multiplier as 7 and average playback rate as the basis for feature. Removing about 2.23% of data elements we obtain:

Similar results with lower correlations are obtained if we remove outliers using this method and remaining feature columns as basis.

* Using Z-score method:

It is observed that if outliers are handled using this method, the resultant correlation has improved strength.

Proof:

Original Correlation:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Feature Name | average\_playback\_rate | total\_DL\_rate | initial\_max\_DL\_rate | 'E2E\_RTT' | initial\_buffering\_latency | playing\_time | stalling\_times | stalling\_ratio | stalling\_duration |
| VMOS | 0.62 | 0.69 | 0.44 | -0.34 | -0.76 | 0.83 | -0.87 | -0.34 | -0.83 |

Correlation after removing outliers:

Taking 3 as max limit for Z-score and average\_playback\_rate feature as basis. Removing 0.99% of data from dataset.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Feature Name | average\_playback\_rate | total\_DL\_rate | initial\_max\_DL\_rate | 'E2E\_RTT' | initial\_buffering\_latency | playing\_time | stalling\_times | stalling\_ratio | stalling\_duration |
| VMOS | 0.67 | 0.74 | 0.44 | -0.34 | -0.76 | 0.83 | -0.87 | -0.34 | -0.83 |

Taking 3 as max limit for Z-score and initial max DL rate feature as basis. Removing 0.99% of data from dataset.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Feature Name | average\_playback\_rate | total\_DL\_rate | initial\_max\_DL\_rate | 'E2E\_RTT' | initial\_buffering\_latency | playing\_time | stalling\_times | stalling\_ratio | stalling\_duration |
| VMOS | 0.62 | 0.79 | 0.44 | -0.34 | -0.76 | 0.83 | -0.87 | -0.34 | -0.83 |

Thus, from above proof it is clear Z-score method is giving more promising result than IQR method. The correlation improvement is significant in case of average playback rate and total DL rate.

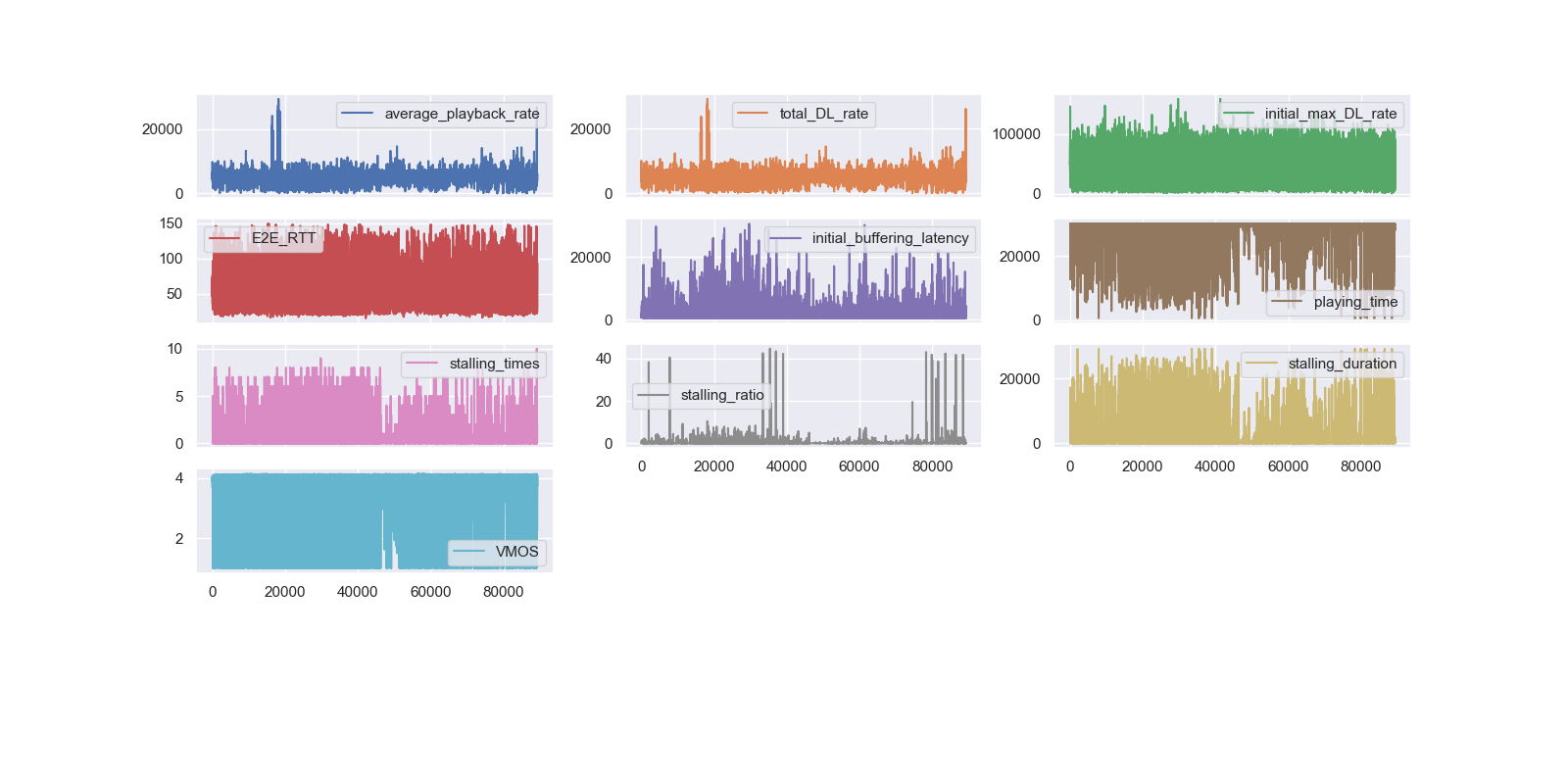
Same results are obtained when total DL rate is used as feature basis.

While for initial max DL rate, the increase is not significant.

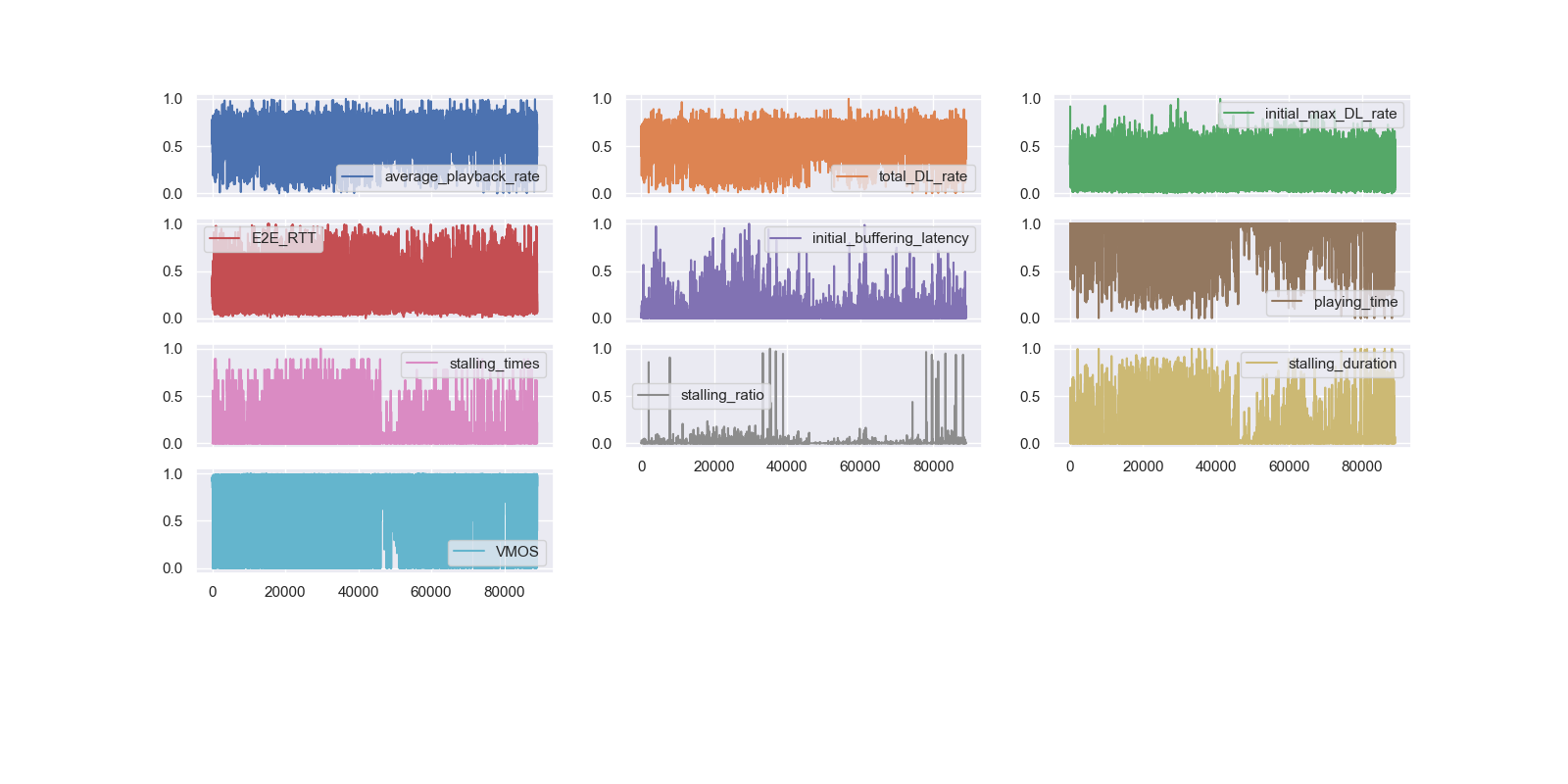
While in case of remaining features there is either no improvement or lowering of strength.

Hence, the optimal way of outlier handling will be using Z-Score method and average playback rate as feature basis.

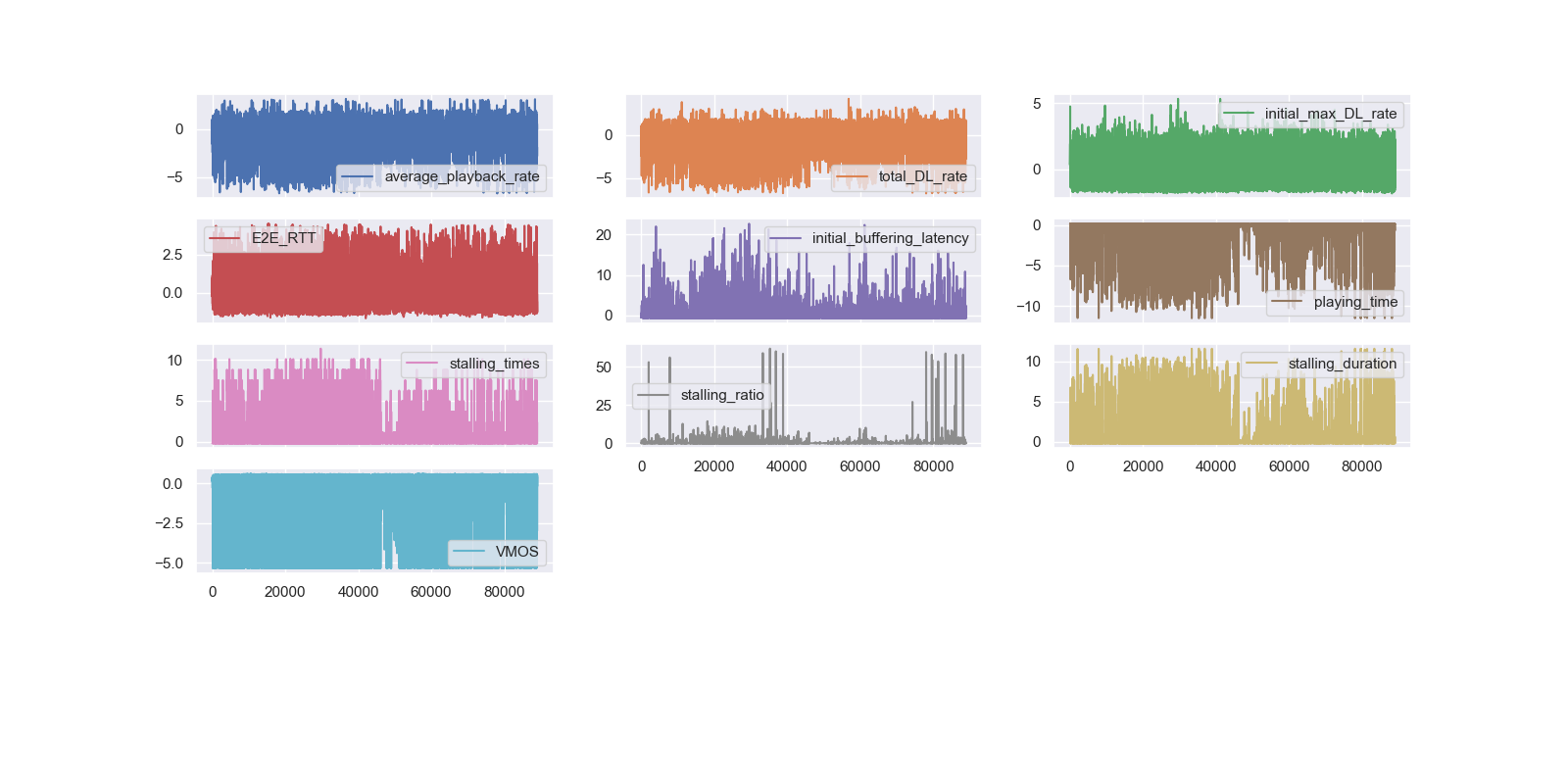
Original Plot:



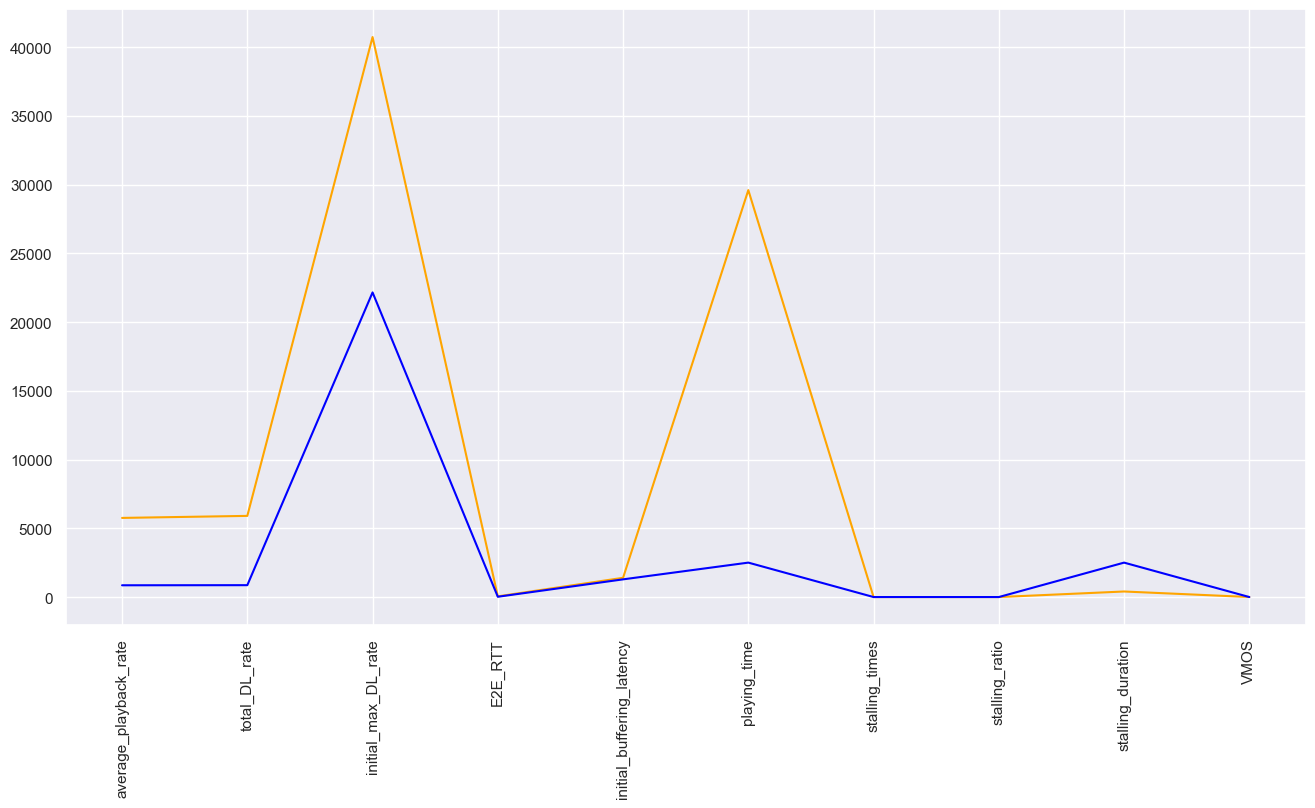
Normalized plot:



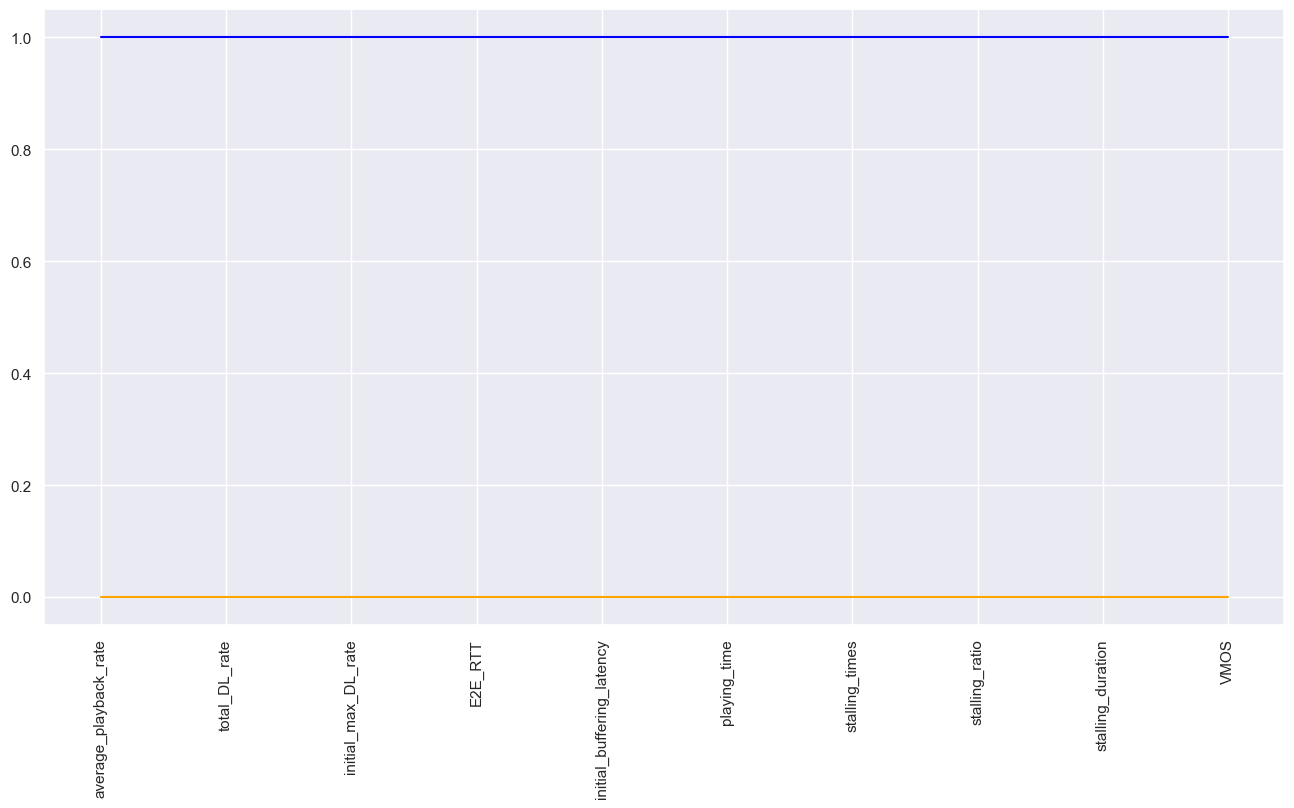
Standardized plot:



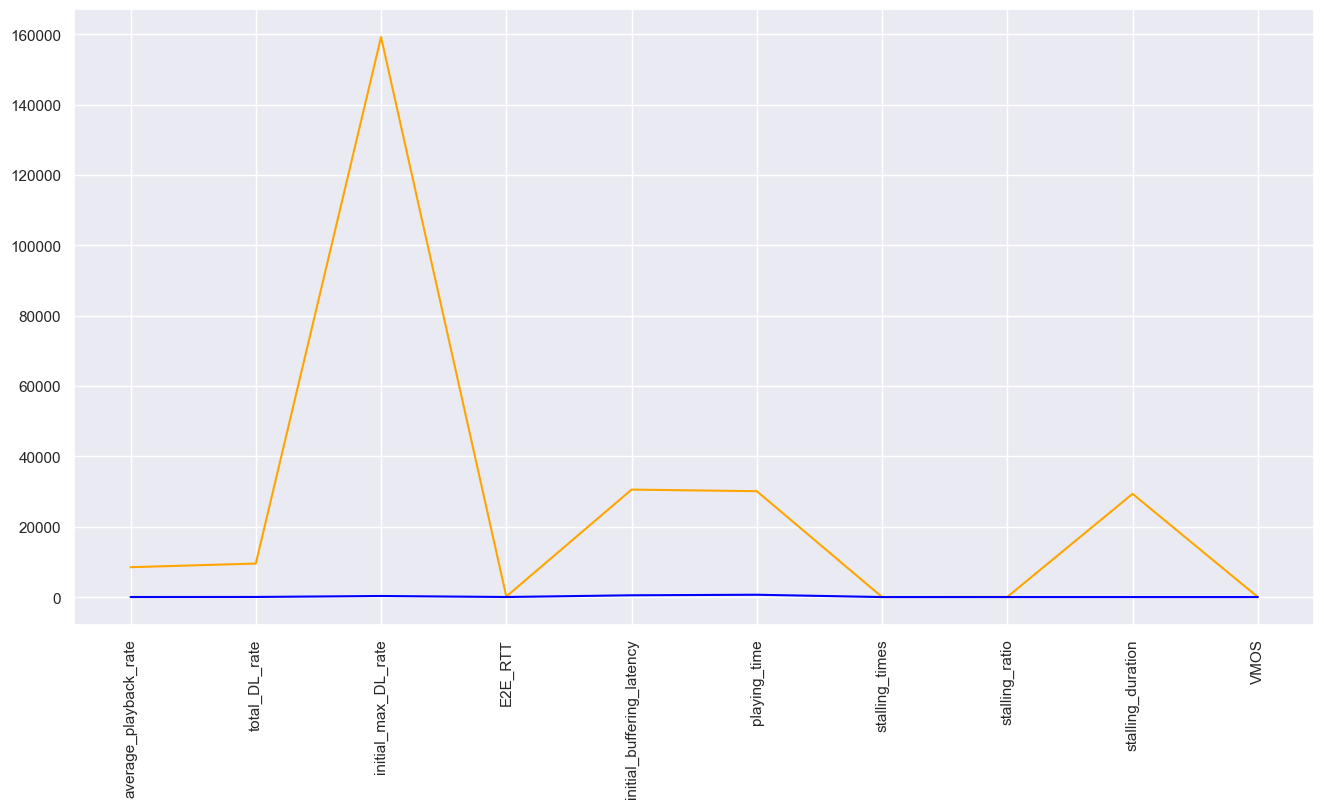
Original Plot: mean std



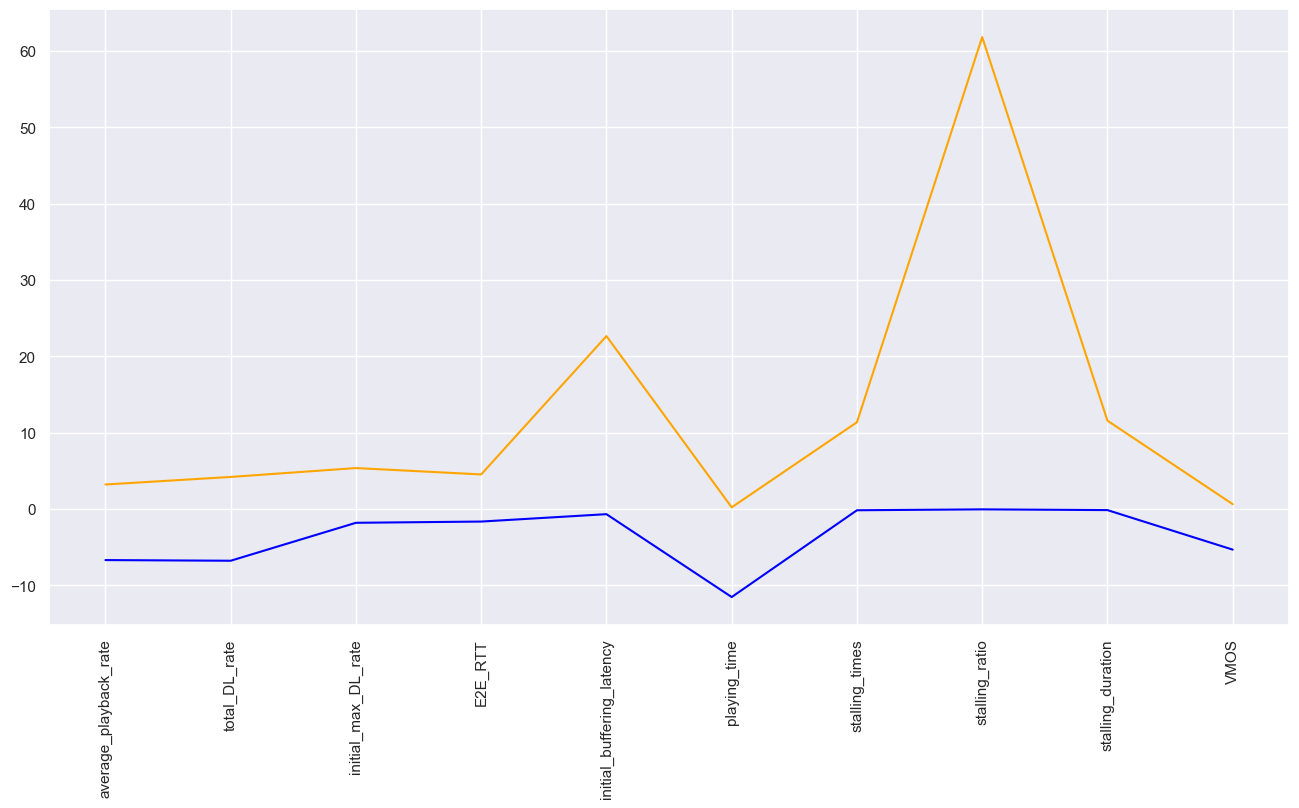
New plot: mean std



Original Plot: maxi mini



New Plot: maxi mini



ACCURACY AND EXECUTION TIME:

PROCESSED DATA:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Number Of PC | ML MODEL | | | | | |
| Logistic Regression  (Multinomial + Saga) | Logistic Regression  (Multinomial + Sag) | Logistic Regression  (OvR + LibLinear) | SVM  (OvR + rbf) | SVM  (OvR + linear) | SVM  (OvR + poly) |
| PC4 | 55.58 s | 32.4 s | 0.5 s | 40 s | 27.7 s | 39.4 s |
| 93.21% | 93.06% | 88.17% | 93.57% | 96.41% | 91.85% |
| PC5 | 36.2 s | 21.5 s | 0.80 s | 29.6 s | 17.90 s | 40.3 s |
| 96.38% | 96.28% | 89.92% | 96.84% | 98.09% | 92.68% |
| PC6 | 37.5 s | 22.2 s | 0.94 s | 20.9 s | 8.58 s | 38.3 s |
| 99.05% | 99.16% | 92.43% | 98.65% | 99.45% | 94.09% |
| PC9 | 49 s | 31.3 s | 1.67 s | 22.23 s | 9.14 s | 44.8 s |
| 99.07 % | 99.18% | 92.97% | 98.83% | 99.59 % | 94.50% |

STANDARDIZED DATA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Logistic Regression  (Multinomial + Saga) | Logistic Regression  (Multinomial + Sag) | Logistic Regression  (OvR + LibLinear) | SVM  (OvR + rbf) | SVM  (OvR + linear) | SVM  (OvR + poly) |
| Time | 48.3 s | 29.7 s | 1.52 s | 22.3 s | 8.78 s | 44.7 s |
| Accuracy | 99.21 % | 99.15 % | 92.79% | 98.67% | 99.43% | 94.53% |

DATA WITHOUT OUTLIERS:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Logistic Regression  (Multinomial + Saga) | Logistic Regression  (Multinomial + Sag) | Logistic Regression  (OvR + LibLinear) | SVM  (OvR + rbf) | SVM  (OvR + linear) | SVM  (OvR + poly) |
| Time | 118 s | 103 s | 1.42 s | 93 s | 66 s | 80 s |
| Accuracy | 99.34 % | 99.45% | 93.77% | 90.36 % | 99.60 % | 90.16% |

RAW DATA:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Logistic Regression  (Multinomial + Saga) | Logistic Regression  (Multinomial + Sag) | Logistic Regression  (OvR + LibLinear) | SVM  (OvR + rbf) | SVM  (OvR + linear) | SVM  (OvR + poly) |
| Time | 287 s | 237 s | 1.48 s | 91 s | 4.59 s | 37 s |
| Accuracy | 96.91% | 97.31% | 97.96 % | 93.08 % | 99.56 % | 94.30% |

Analysis of effect on prediction accuracy using different types of K bins dicretizer:

Accuracy Table using 6 PC’s

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Discretizer | Logistic Regression  (Multinomial + Saga) | Logistic Regression  (Multinomial + Sag) | Logistic Regression  (OvR + LibLinear) | SVM  (OvR + rbf) | SVM  (OvR + linear) | SVM  (OvR + poly) |
| KMeans | 99.05% | 99.16% | 92.43% | 98.65% | 99.45% | 94.09% |
| Uniform | 99.46% | 99.59% | 98.29% | 99.46% | 99.71% | 99.56% |
| Quantile | 94.97% | 95.18% | 77.05% | 95.41% | 97.33% | 79.93% |

Need for discretizing vMOS :

|  |  |  |  |
| --- | --- | --- | --- |
| ACCURACY WITH DICRETE TARGET VARIABLE | | ACCURACY WITH CONTINUOUS TARGET VARIABLE | |
| KMeans + Logistic Regression | KMeans + Support Vector Machine | Linear Regression | Support Vector Regression |
| 99.16% | 99.45% | 87.02% | 77.77% |