EP2300 Project - Network Management Estimating Conformance to Service Level Agreements (SLAs) using Machine Learning

Jiani Jiang <jianij@kth.se>

Task I - Data Exploration:

1. Compute the following statistics for each feature of X and target of Y: mean, maximum, minimum, 25th percentile, 90th percentile, and standard deviation. Give no more than two digits after decimal point.

Answer:

Mean values of X and Y:

```
--------------------------------------------------------------------------------
Mean value of X: plist-sz     8.76e+02
totsck        4.85e+02
ldavg-1       7.33e+01
pgfree/s      1.50e+05
proc/s        8.00e+00
all_%%usr     8.62e+01
file-nr       2.58e+03
cswch/s       5.25e+04
%%memused     1.33e+01
runq-sz       6.34e+01
TimeStamp     1.41e+09
dtype: float64
Mean value of Y: Unnamed: 0     1.80e+03
DispFrames     1.89e+01
TimeStamp      1.41e+09
dtype: float64
--------------------------------------------------------------------------------
```

Maximum values of X and Y:

```
--------------------------------------------------------------------------------
Maximum value of X: plist-sz     1.41e+03
totsck        7.44e+02
ldavg-1       1.56e+02
pgfree/s      8.65e+05
proc/s        5.80e+01
all_%%usr     9.81e+01
file-nr       2.98e+03
cswch/s       8.47e+04
%%memused     1.76e+01
runq-sz       1.50e+02
TimeStamp     1.41e+09
dtype: float64
Maximum value of Y: Unnamed: 0     3.60e+03
DispFrames     3.02e+01
TimeStamp      1.41e+09
dtype: float64
--------------------------------------------------------------------------------
```

Minimum values of X and Y:

```
--------------------------------------------------------------------------------
Minimum value of X: plist-sz     4.04e+02
totsck       2.41e+02
ldavg-1      1.79e+00
pgfree/s     1.80e+02
proc/s       0.00e+00
all_%%usr    1.96e+00
file-nr      2.11e+03
cswch/s      2.73e+03
%%memused    6.19e+00
runq-sz      0.00e+00
TimeStamp    1.41e+09
dtype: float64
Minimum value of Y: Unnamed: 0    0.00e+00
DispFrames    0.00e+00
TimeStamp     1.41e+09
dtype: float64
--------------------------------------------------------------------------------
```

25th percentile value of X and Y:

```
--------------------------------------------------------------------------------
25th percentile value of X: plist-sz     6.11e+02
totsck       3.54e+02
ldavg-1      2.06e+01
pgfree/s     1.32e+05
proc/s       0.00e+00
all_%%usr    7.47e+01
file-nr      2.40e+03
cswch/s      2.94e+04
%%memused    1.22e+01
runq-sz      2.10e+01
TimeStamp    1.41e+09
Name: 0.25, dtype: float64
25th percentile value of Y: Unnamed: 0    9.00e+02
DispFrames    1.34e+01
TimeStamp     1.41e+09
Name: 0.25, dtype: float64
--------------------------------------------------------------------------------
```

90th percentile value of X and Y:

```
--------------------------------------------------------------------------------
90th percentile value of X: plist-sz     1.25e+03
totsck       6.72e+02
ldavg-1      1.36e+02
pgfree/s     1.86e+05
proc/s       2.10e+01
all_%%usr    9.75e+01
file-nr      2.83e+03
cswch/s      7.23e+04
%%memused    1.58e+01
runq-sz      1.11e+02
TimeStamp    1.41e+09
Name: 0.9, dtype: float64
90th percentile value of Y: Unnamed: 0    3.24e+03
DispFrames    2.40e+01
TimeStamp     1.41e+09
Name: 0.9, dtype: float64
--------------------------------------------------------------------------------
```

standard deviation value of X and Y:

```
--------------------------------------------------------------------------------
standard deviation value of X: plist-sz       267.28
totsck          132.24
ldavg-1          48.20
pgfree/s      30798.85
proc/s            9.03
all_%%usr        18.15
file-nr         184.89
cswch/s       20576.24
%%memused         1.86
runq-sz          37.08
TimeStamp      1039.37
dtype: float64
standard deviation value of Y: Unnamed: 0    1039.37
DispFrames        5.46
TimeStamp      1039.37
dtype: float64
--------------------------------------------------------------------------------
```

2. Compute the following quantities of X:

(a) The number of observations with CPU utilization ("all %%usr") smaller than 90% and memory utilization ("%%memused") smaller than 50%; (b) The average number of used sockets ("totsck") for observations with less than 60000 context switches per seconds ("cswch/s").

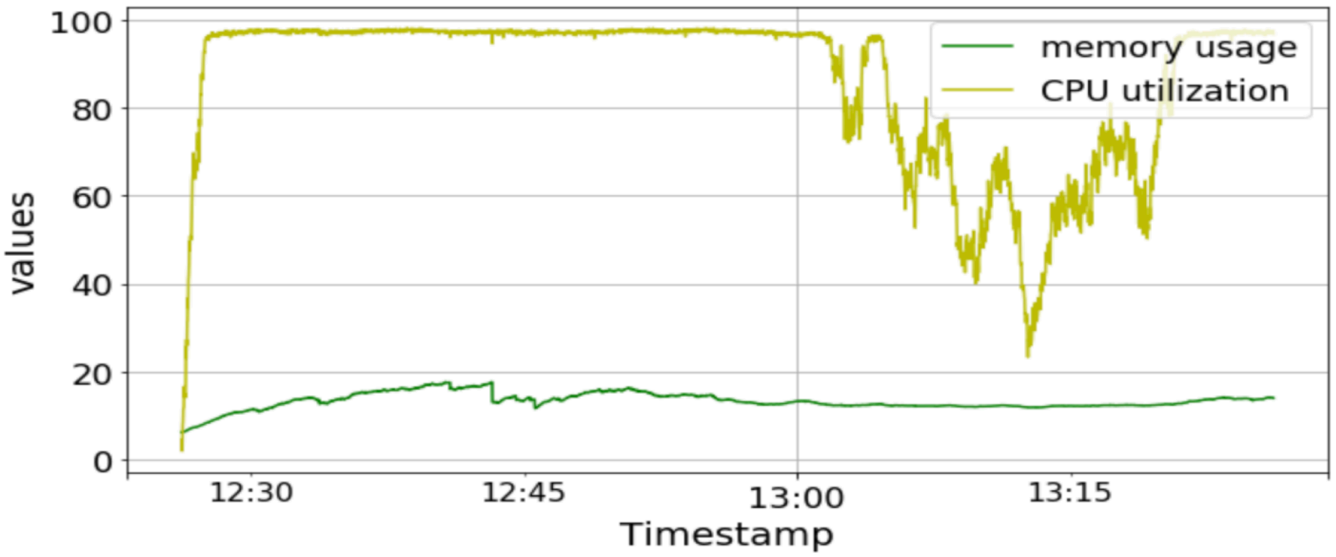
Answer:

```
The number of observations with CPU utilization ('all_%%usr') smaller than 90% and memory utilization ('%%memused') s
maller than 50%: 1114
The average number of used sockets ('totsck') for observations with less than 60000 context switches per seconds ('cs
wch/s'): 356.12
```

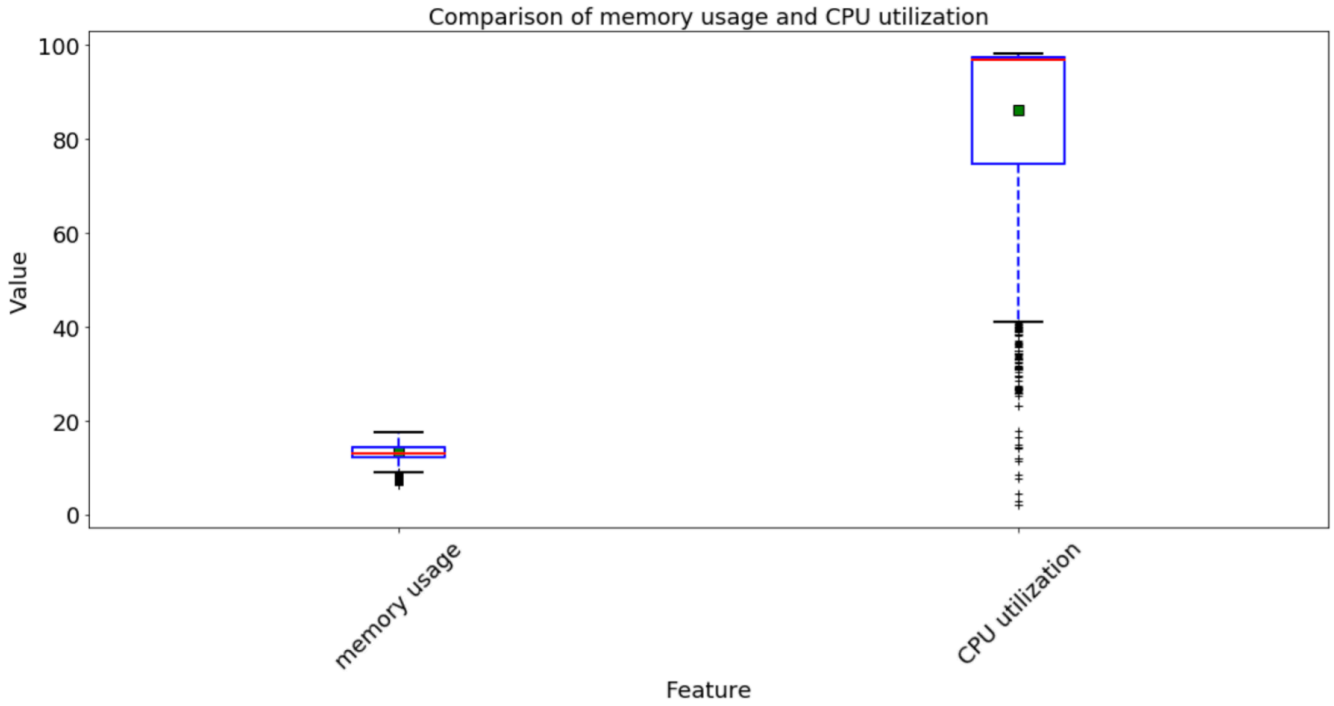
3. Produce the following plots: (a) Time series of memory usage ("%%memused") and CPU utilization ("all %%usr"), both curves in a single plot. Box plot of both features in a single plot. (b) Density plots of memory usage ("%%memused") and CPU utilization ("all %%usr"), Histograms of both these features (choose a bin size of 1%), four plots in all.

Answer:

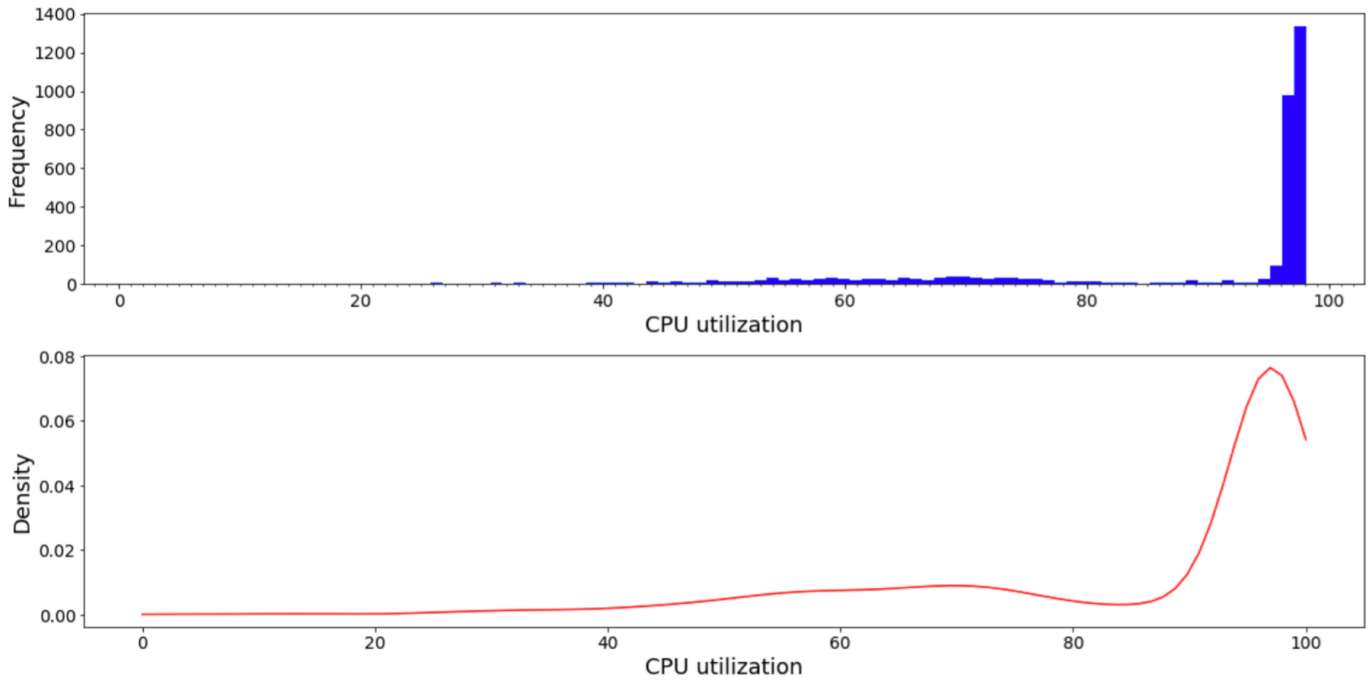
(a) Time series of memory usage ("%%memused") and CPU utilization ("all_%%usr"):



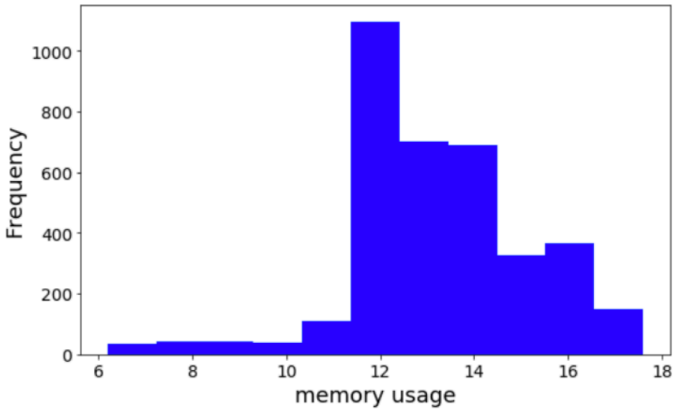
Box plot of both features in a single plot:

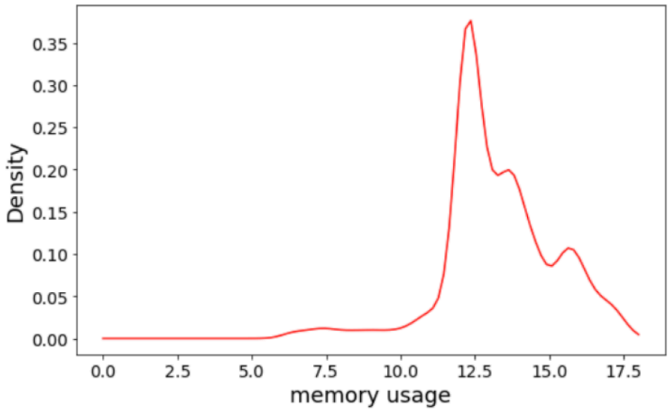


(b) Density plot and histogram plot of CPU utilization



Density plot and histogram plot of memory usage:





Task II - Estimating Service Metrics from Device Statistics:

1. Evaluate the Accuracy of Service Metric Estimation.

Answer:

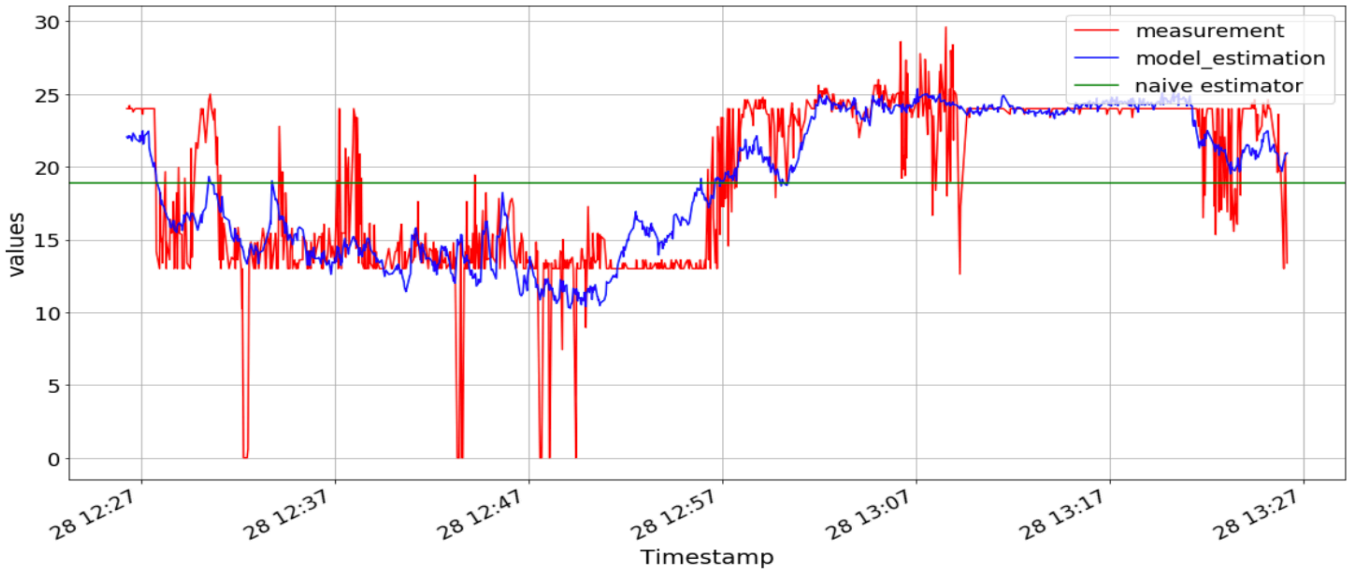
(a)

```
Coefficients: [-2.10e-02  1.20e-02  3.34e-03 -6.00e-06 -8.52e-03  9.30e-02 -2.98e-03
 -8.70e-05  4.10e-01 -1.39e-02]
Intercept: 31.801
```

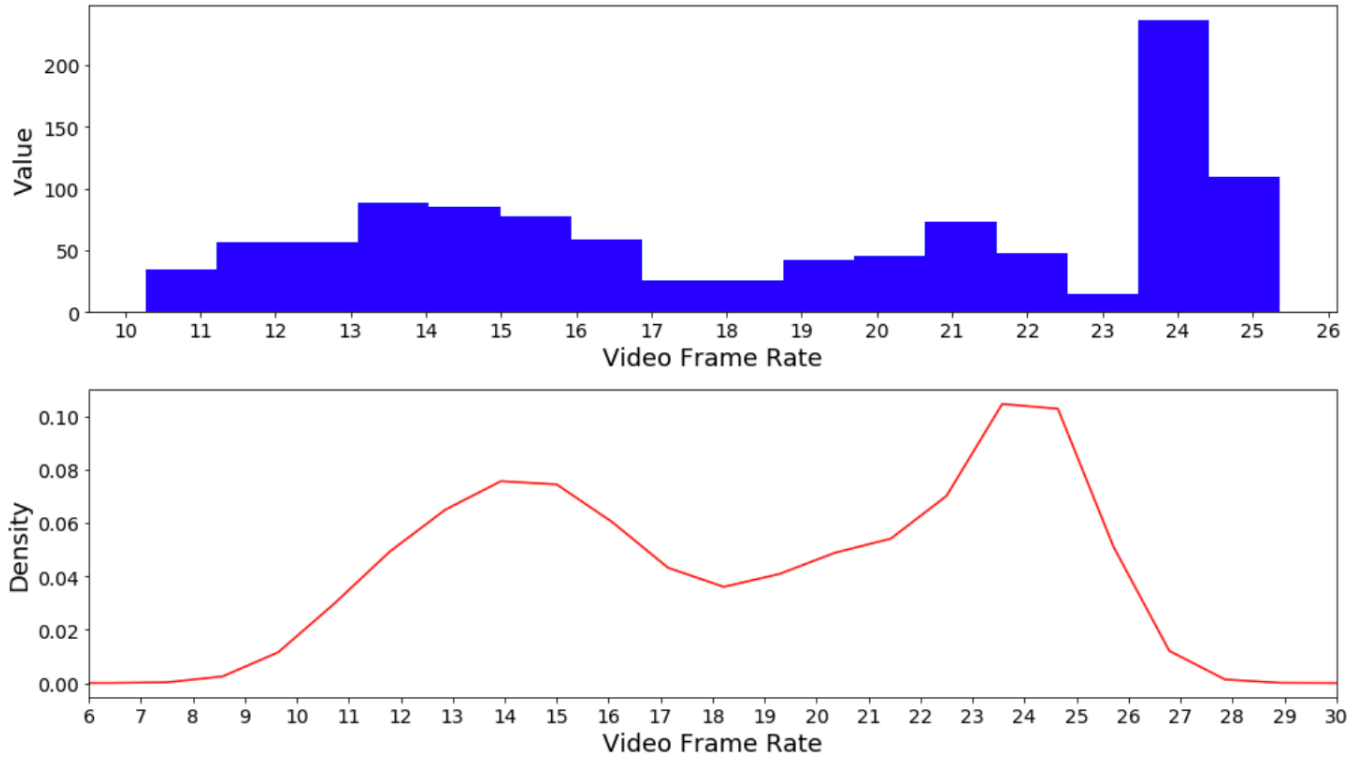
(b)

```
NMAE for Linear Regression: 0.100
Mean value of y for the naive method for the training set: 18.8
NMAE for Naive method: 0.255
```

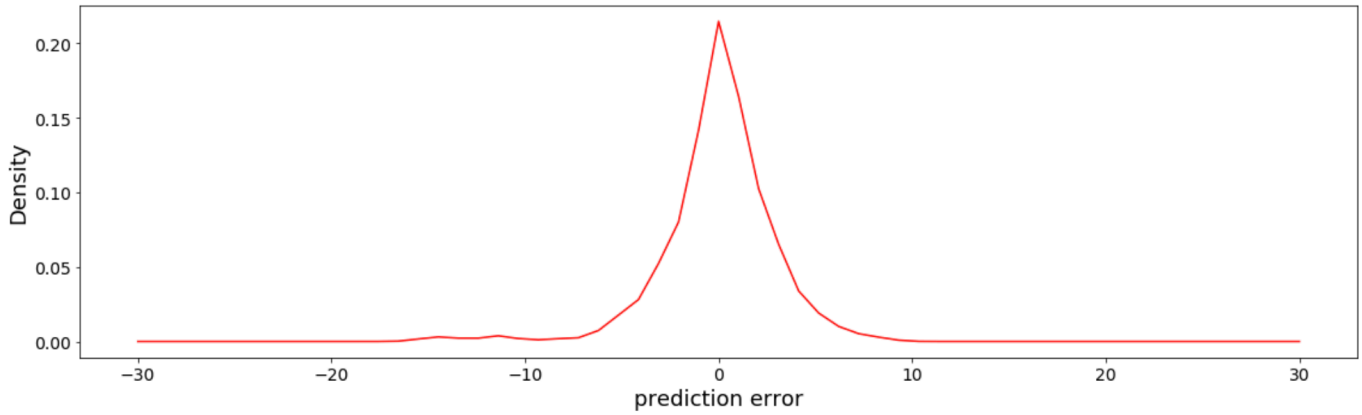
(c) The time series plot of measurements and model estimations for M for Video Frame Rate:



(d) The density and histogram plots for Video Frame Rate in test set:



(e) The density plot for the prediction error:



(f) In this module, I use ten features to estimate the Video Frame Rate, and get 10 coefficients and one bias using linear regression to train a model M with the training set. Also, I use naïve method to simply predict target variables in test set of Y that only rely on the train set of Y. The naïve estimator here is 18.9 and the first plot shows that the red curve (the trend of Y_test) is basically around the green line (the naïve estimator).

The figures of NMAE directly shows that for the test set, the NMAE of Linear Regression is 0.101, while the NMAE of naïve method is 0.265. A lower NMAE value means a better model. Linear Regression has a better accuracy than naïve method for this model.

The density and histogram plots both reflect the data distribution of video frame rate, and they are mostly consistent here. They show that video frame rate has two peaks at 14 frame/s and 24 frame/s, so, more predicted video frame rates are distributed at two intervals of 14 frame/s and 24 frames/s. And it reaches the maximum number of 220 (around 0.105 of the density) when the rate is 24 frame/s. Meanwhile, the value turns to 0 when a frame rate is bigger than 26 or smaller than 10, and the curve of density plot tend to be smooth in this two intervals as well.

The density plot for prediction error illustrates that there is only one peak appears at interval [-8,8] and reach the maximum value of 0.22 at point 0, which illustrates that almost all predicted rates have the error that less than 8 frame/s compared to measured rates, and most of them have extremely small errors that close to 0.

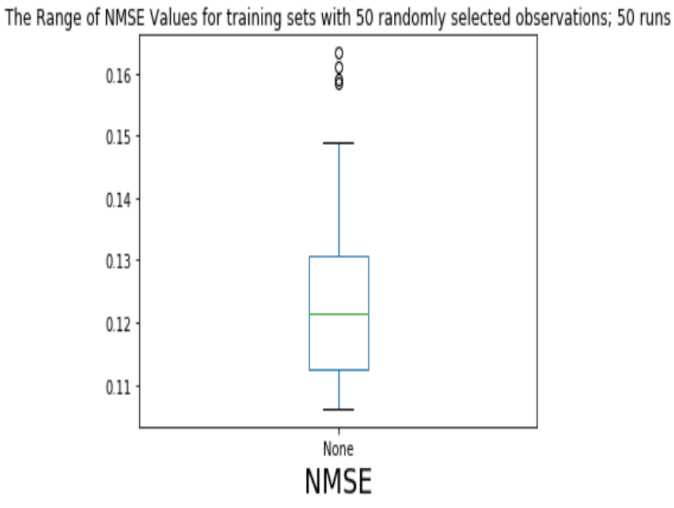
2. Study the Relationship between Estimation Accuracy and the Size of the Training Set.

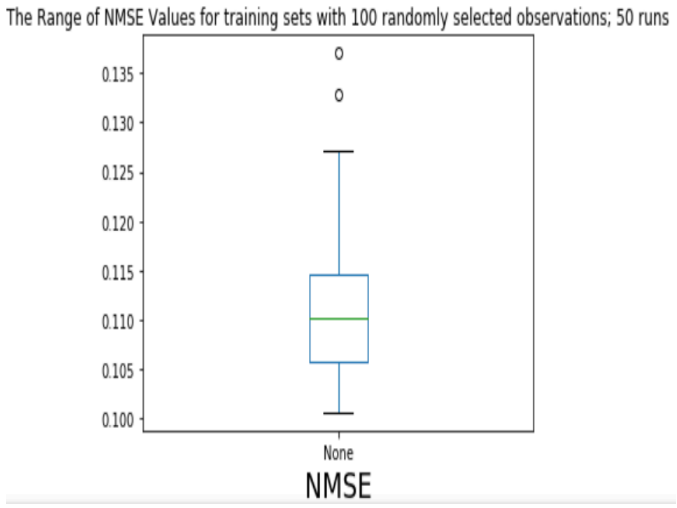
Answer:

(a)

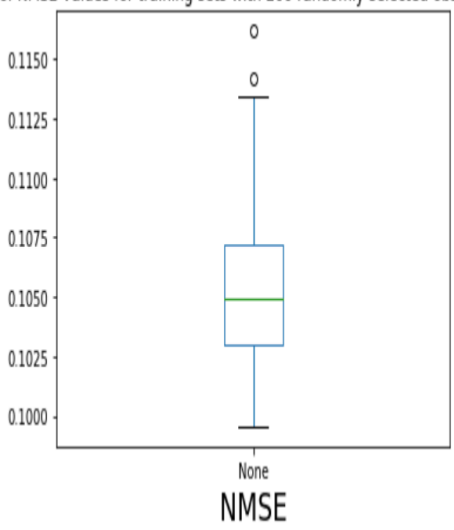
```
Coefficients of randomly choose 50 observations from the training set with 2520 observation: [-7.09e-02  7.49e-02  1.
19e-01  2.62e-05 -3.98e-02  6.14e-02  1.85e-03
 -1.36e-04  9.31e-01 -2.13e-02]
Intercept:18.119
NMAE for the test set: 0.12165
==========================================================================
Coefficients of randomly choose 100 observations from the training set with 2520 observation: [ 2.07e-02 -1.05e-01
6.36e-02  2.25e-05  1.18e-02  1.83e-02 -4.84e-03
 -1.61e-05  2.70e-01  2.35e-02]
Intercept:50.273
NMAE for the test set: 0.10884
==========================================================================
Coefficients of randomly choose 200 observations from the training set with 2520 observation: [ 7.51e-03 -4.05e-02 -
2.57e-02 -9.58e-06  7.59e-03  1.17e-01 -7.42e-04
 -9.56e-05  9.26e-01 -3.57e-02]
Intercept:21.659
NMAE for the test set: 0.10488
==========================================================================
Coefficients of randomly choose 500 observations from the training set with 2520 observation: [-2.29e-02  1.46e-02
1.84e-02 -8.32e-06 -6.99e-03  9.46e-02 -3.30e-03
 -8.96e-05  2.05e-01 -1.53e-02]
Intercept:35.157
NMAE for the test set: 0.10135
==========================================================================
Coefficients of randomly choose 1000 observations from the training set with 2520 observation: [-2.14e-02  1.18e-02
8.54e-03 -3.41e-06 -6.66e-03  9.04e-02 -2.34e-03
 -8.90e-05  4.55e-01 -2.04e-02]
Intercept:29.971
NMAE for the test set: 0.10050
==========================================================================
Coefficients of randomly choose 2520 observations from the training set with 2520 observation: [-2.10e-02  1.20e-02
3.34e-03 -6.00e-06 -8.52e-03  9.30e-02 -2.98e-03
 -8.70e-05  4.10e-01 -1.39e-02]
Intercept:31.801
NMAE for the test set: 0.09995
```

(d)

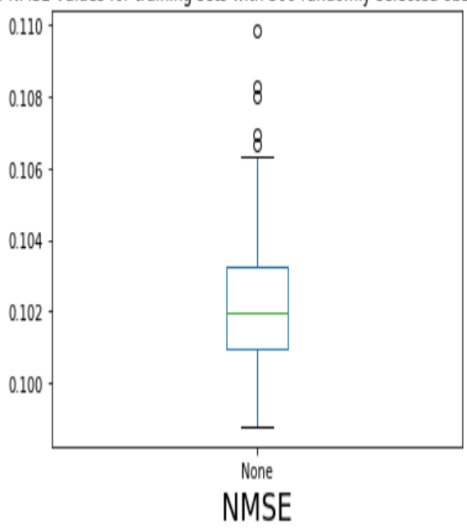




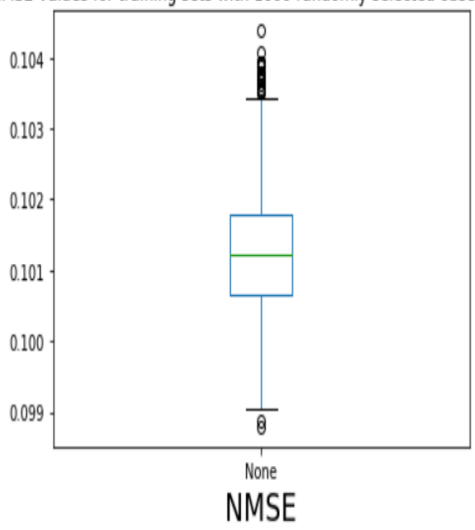
The Range of NMSE Values for training sets with 200 randomly selected observations; 50 runs



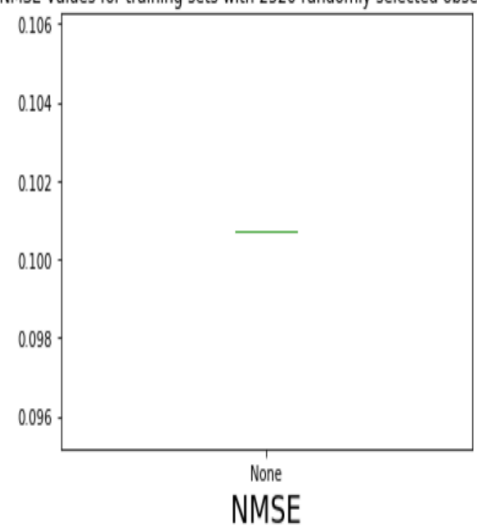
The range of NMSE Values for training sets with 500 randomly selected observations; 50 runs



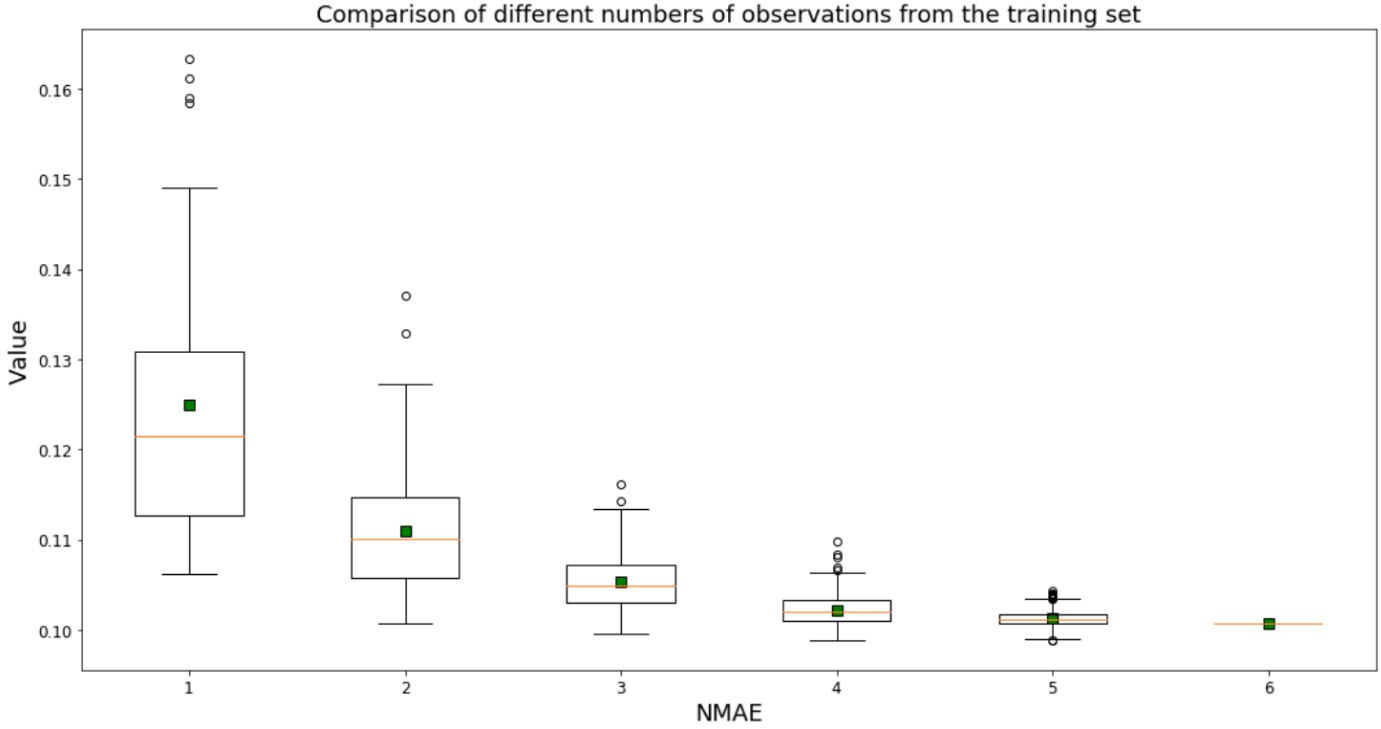
Range of NMSE Values for training sets with 1000 randomly selected observations; 50 runs



Range of NMSE Values for training sets with 2520 randomly selected observations; 50 runs

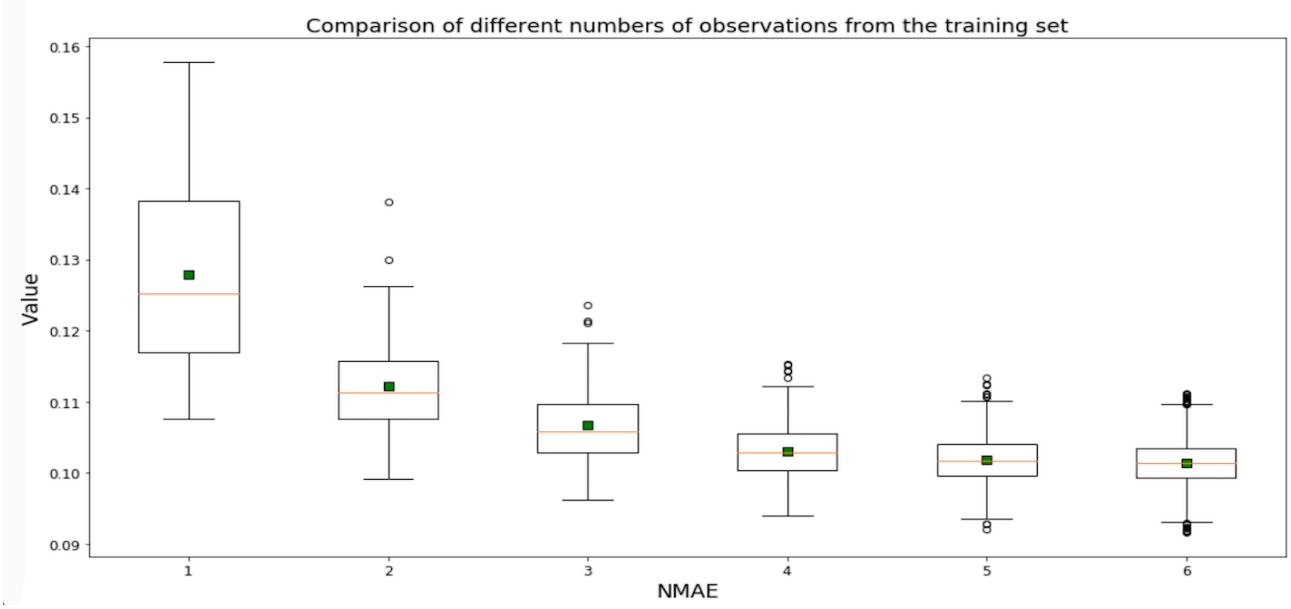


Comparison of different numbers of observations from the training set



(e) In this task, I creat another six training sets from the original one to test the relationship between the size of the training set and estimation accuracy. The figures of NMAE shows that, as the size of training set grows, the value of NMAE decreases and accurancy increased.
To avoid contigency, I train linear models and compute the NMAE for each model above 50 times, and use box plots to show the range of the NMAE values for each given set size. As the size grows, the green point which represents the mean value of MNAE shows a downward trend and it means the accuracy is improving. Meanwhile, the distance between upper quartile and lower quartile is smaller with the training size increasing. The smaller the distance, the more concentrated it is. And it's the same for upper limb and lower limb. The statistics lead us to the conclusion that, as the size of training set increases, the results of NMAE are more concentrated and the estimation accuracy is better.

PS: I considered another way to train models and perform 50 times: Every time before I select the new set from the training set(e.g.100), I first re-split the whole set (to 70% and 30%) to training set and test set again. So this means I re-split the original set to 70% and 30% 50 times as well. The result is :



It has the same pattern that, when we increase the size of train set, the values of MNAE are more concentrated and the mean value of MNAE becomes smaller. But I find that, with re-spliting the data set, the errors are more random. Because when I split the data set, the test set will be different. Some test sets will gain good results for some models and bad for others. So, it's not fair to use different test sets for comparison. To evaluate the models, it's better to compare the results using same data for evaluation.
Also, I observed that, it has a slower convergence than without re-spliting data set method. I think the reason is randomness of new selected set and test set largely increased. And the uncertainty of errors also contributes to this result. Due to the reason of re-spliting data set, it can never converge to a straight line as previous one.

Task III - Estimating SLA Conformance and Violation from Device Statistics:

1.Model Training - use Logistic Regression to train a classifier C with the training set. Provide the coefficients of your model C. Give no more than three significant digits.

Answer:

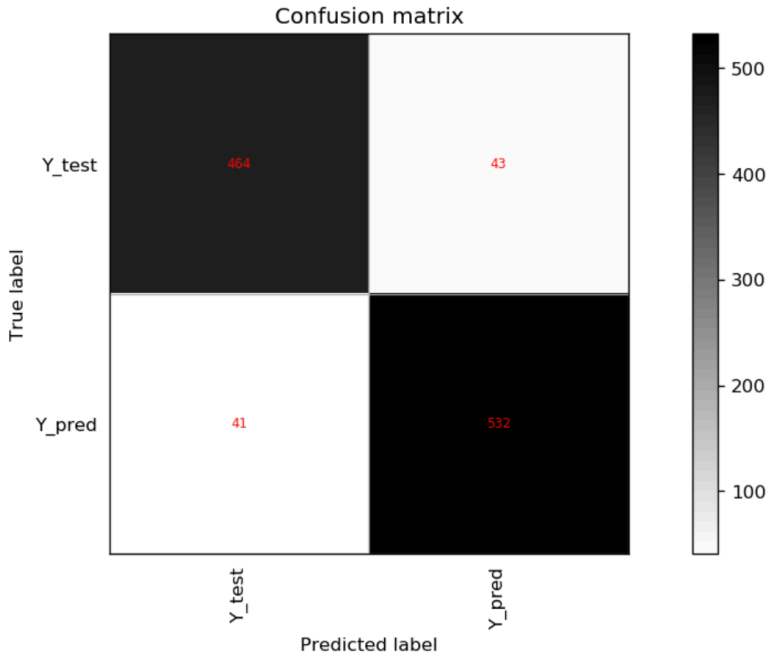
The coefficients of my model C:

```
Coefficients: [[-1.07e-02  7.67e-04 -1.14e-02  9.31e-07 -1.39e-03  2.68e-02  4.33e-03
  -3.98e-05  7.03e-03 -2.37e-02]]
```

2.Accuracy of the Classifiers C - Compute the classification error (ERR) on the test set for C. For this, you first compute the confusion matrix, which includes the four numbers True Positives (TP), True Negatives (TN), False Positives (FN), and False Negatives (FN). A true positive is an observation that is correctly classified by the classifier as conforming to the SLA; a true negative is an observation that is correctly classified by the classifier as violating the SLA. Use confusion matrix plot to show TP, TN, FP and FN.

Answer:

Confusion matrix plot:



TP, TN, FP and FN:

```
[[464  43]
 [ 41 532]]
```

ERR = 0.078

3.As a baseline for C, use a naive method which relies on Y values only, as follows. For each x belongs to X, the naive classifier predicts a value True with probability p and False with probability 1 - p. p is the fraction of Y values that conform with the SLA. Compute p on the training set and the classification error for the naive classifier on the test set.

Answer:

Naive method:

```
naive classifier p = 0.533
ERR = 0.498
```

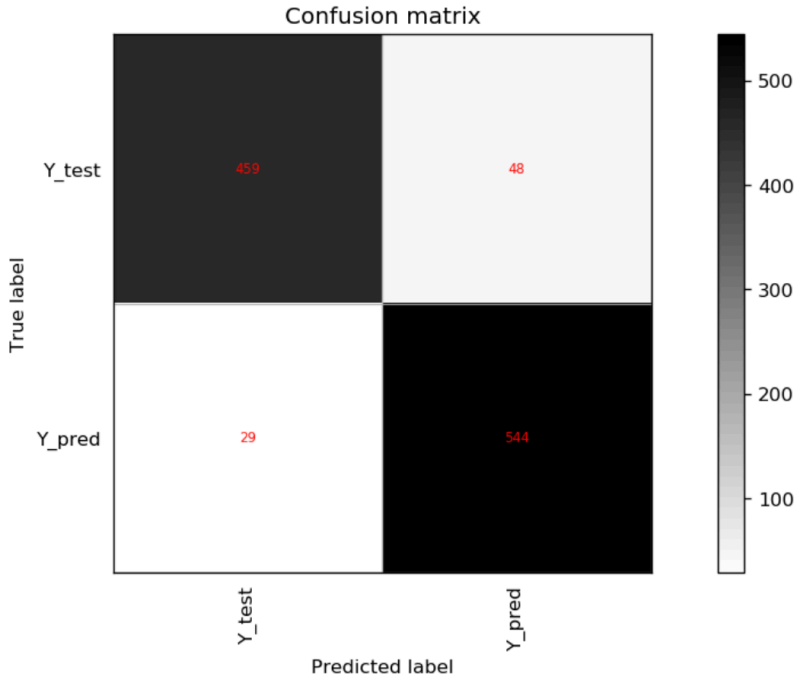
4.Build a new classifier by extending extend the linear regression function developed in Task II with a check on the output, i.e., the Video Frame Rate. If the frame rate for a given X is above the SLA threshold, then the Y label of the classifier is set to conformance, otherwise to violation. Compute the new classifier on the training set and the classification error for this new classifier on the test set.

Answer:

The new classifier by extending extend the linear regression function:

```
Coefficients: [-1.36e-02 -1.71e-03  6.50e-03 -4.60e-06 -3.67e-03  8.65e-02 -3.39e-03
 -8.52e-05  2.98e-01 -1.40e-02]
```

Confusion matrix plot:



TP, TN, FP and FN:

```
[[459  48]
 [ 29 544]]
```

ERR = 0.071

5.Formulate your observations and conclusions based on the above work.

Answer:

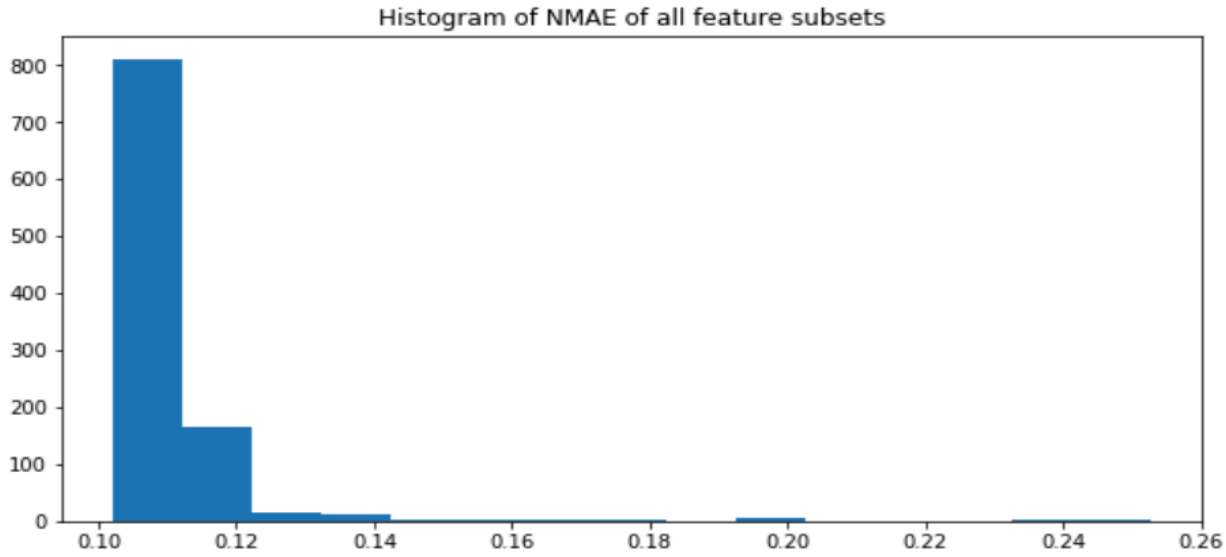
In my case, the ERRs for Logistic Regression is 0.078 and for Linear Regression is 0.071. We can find that the ERR for Linear Regression is much the same as Logistic Regression. Which means that Logistic Regression and Linear Regression have the same accuracy. However, the naïve method has the worst accuracy with the ERR equals to 0.498.

In my last version of the code, I made two mistakes. One is that when I added the new column “low rate” to Y set, I compared the column “DispFrames” with 18. At first, I set the item of its value smaller than 18 to True, and False if it’s larger than 18. I get the number of the Y that violates the SLA as 1 which is the complete opposite. I should set the item to True when its value larger than 18, which conforms to the SLA. Because later we calculate TP, which is correctly classified by the classifier as conforming to the SLA. Also, TN, FP and

FN have corresponding meanings. So we need to first set the values that equal and larger than 18 to be True. I messed the meaning of conformation and violation.

The second mistake I made is forget to set the parameter C of Logistic Regression. C is the Inverse of regularization strength and default to be 1. I need to set it as a big number of 1e5, which is 100000, to have the minimize regularization in order to avoid overfitting.

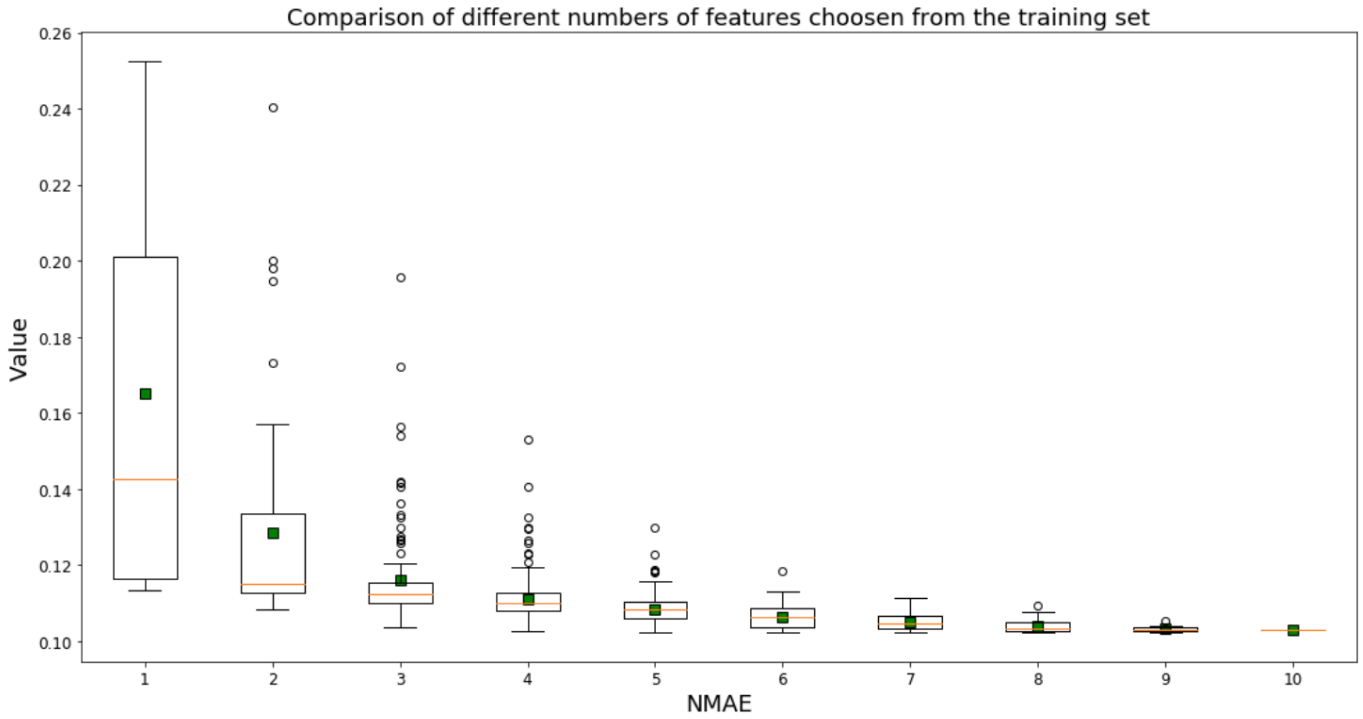
Task IV - Reduce the Number of Device Statistics to Estimate the Service Metric:
1.Method 1 (Optimal method; this method is appropriate if X contains a small number of features): Build all subsets of the feature set X. Compute a linear regression model for each of these subsets of the training set. For each model compute the error (NMAE) on the test set. Draw histograms of these errors. In addition, list the device statistics (i.e., features) of the model with the smallest error. Produce a plot that contains 10 box plots, one box plot each with the errors (NMAE) for all models that contain 1 feature, 2 features, ..., 10 features.
Answer:
Histogram of these errors of all feature subsets:



List the device statistics (i.e., features) of the model with the smallest error:

```
The smallest erorr is : 0.1022070035265209
('plist-sz', 'pgfree/s', 'proc/s', 'all_%%usr', '%%memused', 'runq-sz')
```

The plot that contains 10 box plots, with the errors (NMAE) for all subsets of features:

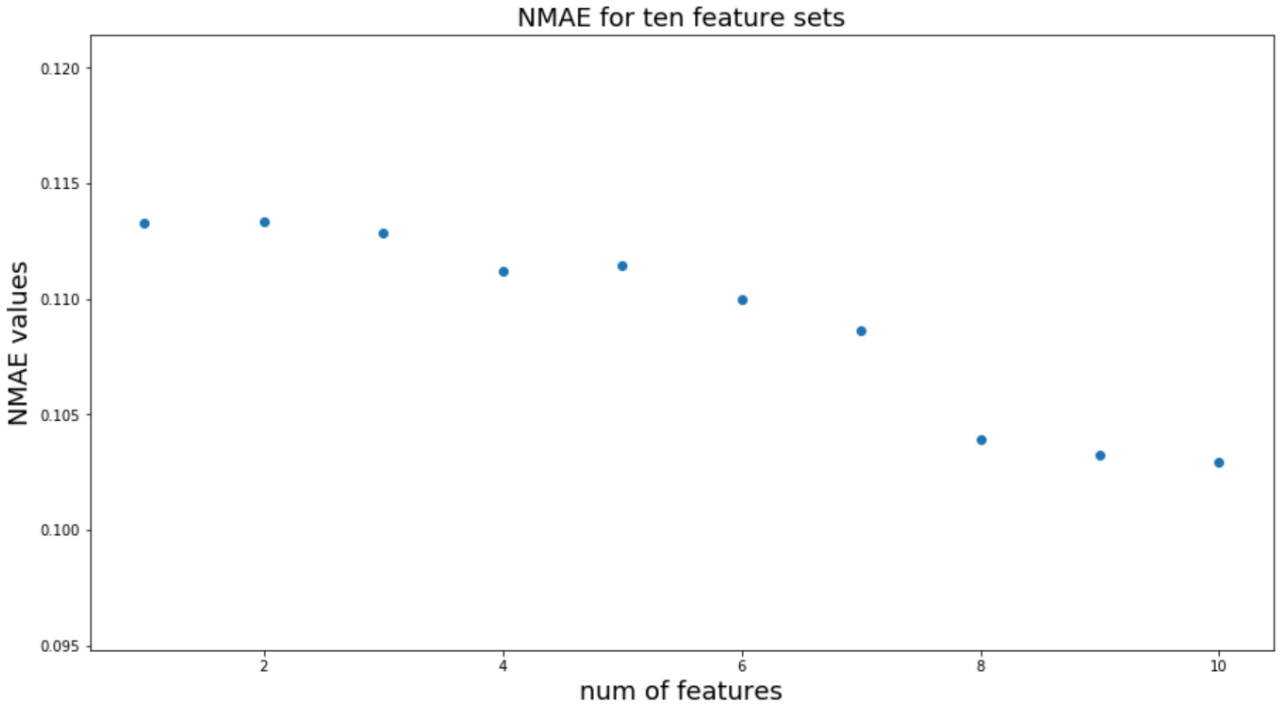


2.Method 2 (Heuristic method): Linear univariate feature selection. Take each feature of X and compute the sample correlation of the feature with the Y value on the training set. For feature x and target y, the sample correlation is computed as $\frac{1}{m}\sum_{i=1}^{m}(x_i - \bar{x})(y_i - \bar{y})/(\sigma_X - \sigma_Y)$. The correlation values fall into the interval [1, +1]. Rank the features according to the square of the correlation values; the top feature has the highest value. Build ten feature sets composed of the top k features, k = 1, ..., 10. The first feature set contains the top feature, the second feature set contains the top two features, etc. For each feature set, compute the linear regression model on the training set and compute the error (NMAE) on the test set. Produce a plot that shows the error value in function of the set size k.
Plot a heat map of the correlation matrix whose vertical axis and horizontal axis is the same vector (x1, x2, ..., x10, y).

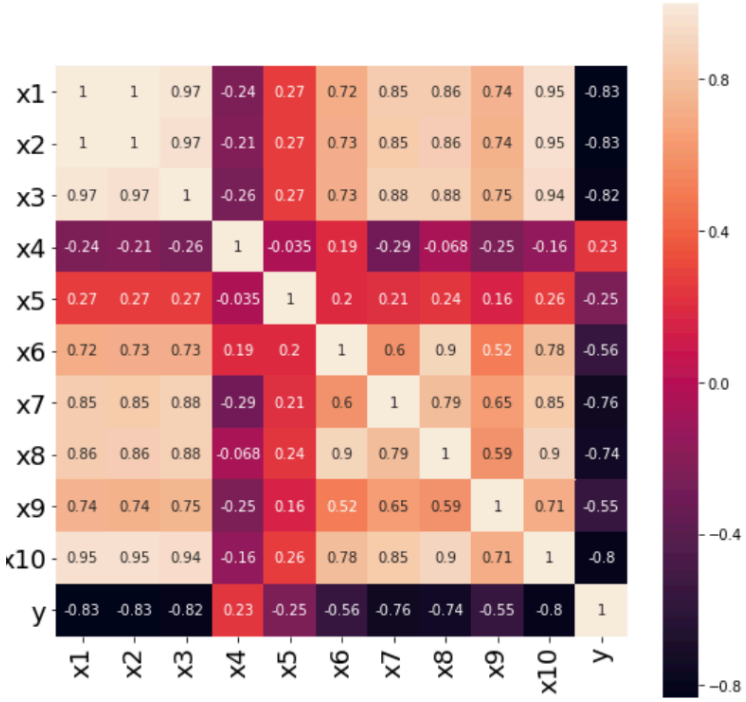
Answer:

The plot that shows the error value in function of the set size k:

```
NMAE for ten feature sets:  [0.1133006452269017, 0.11332893054537599, 0.11286597780032097, 0.1111879
3014838257, 0.11143163035513387, 0.10996587809031623, 0.10863076611433656, 0.10391297757818295, 0.10
323223518896575, 0.10293012530047824]
```



The heat map of the correlation matrix:



3.Compare these three methods against each other and explain the advantages and disadvantages of each method in terms of required computing time and achieved accuracy. Which method produces a small feature set whose model has an error very close to the smallest possible error?

Answer:

The advantage of Optimal method is that it calculates all combinations of features, so that it can get a best option with the smallest error. Compared to Heuristic, it has better accuracy. The disadvantage of Optimal

method is it takes a long time to complete the execution. For example, the computing time for Optimal method is 6.544 second and for Heuristic method is 0.830 second in one case.
On the contrary, Heuristic method doing less work by first sorting features according to their correlation values. It's execution speed is extremely fast and saves a lot of time. However, it is hard to find the perfect combination of features that has the smallest error. So its accuracy is not as good as Optimal method.
In a word, Optimal method can provide a smaller feature set that has an error very close to the smallest possible error while is time-consuming.