Project 1, Task II: Estimating Service Metrics from Device Measurements

Yiyi Miao

November 7, 2025

Abstract

This report addresses Task II of Project 1, focusing on estimating service-level metrics from device statistics for two services: a Key-Value (KV) store and a Video-on-Demand (VoD) service. Using the reduced feature matrix X' (18 features) from Task I, we train and evaluate three regression models: M_1 (Linear Regression), M_2 (Random Forest Regression), and M_3 (Neural Network Regression). We compare their performance based on the Normalized Mean Absolute Error (NMAE) and computational overhead (training time). We also establish a Naïve Baseline for comparison.

1 Model Training and Data Split

The data used for this task is the reduced design matrix X' (containing the top 18 features identified in Task I) and the corresponding target vector Y. The target for the KV dataset is `ReadsAvg`, and for the VoD dataset, it is `DispFrames`.

As specified, the dataset for each service was split into a training set (70% of samples) and a test set (30% of samples), selected uniformly at random with a `random_state` of 42 for reproducibility.

The resulting dataset sizes are as follows:

- **KV Dataset:** 6805 training samples and 2917 test samples.
- **VoD Dataset:** 12821 training samples and 5496 test samples.

2 Model Accuracy and Training Time

The accuracy of each model was evaluated on the 30% test set using the Normalized Mean Absolute Error (NMAE), defined as:

$$\text{NMAE} = \frac{1}{\overline{y}}\left(\frac{1}{m}\sum_{j=1}^{m}|y_j - \hat{y}_j|\right)$$

where $\overline{y}$ is the mean of the test set target values, y_j is the measured value, and $\hat{y}_j$ is the model's prediction. The computational overhead is measured as the model's training time in seconds.

2.1 KV Service (ReadsAvg)

Table 1 summarizes the performance of the trained models on the KV dataset. The Random Forest (M_2) model achieved the lowest NMAE (0.019), indicating the highest accuracy. The Linear Regression (M_1) model was the fastest to train, while the Neural Network (M_3) was by far the most computationally expensive due to the hyperparameter search. All three machine learning models significantly outperformed the Naïve Baseline.

Table 1: Model Comparison for KV Dataset (Target: ReadsAvg)

Model	**NMAE**	**Training Time (s)**
M_1 (Linear Regression)	0.021	0.015
M_2 (Random Forest)	0.019	1.024
M_3 (Neural Network)	0.023	335.278
Naïve Baseline	0.041	N/A

2.2 VoD Service (DispFrames)

Table 2 summarizes the performance on the VoD dataset. Similar to the KV results, the Random Forest (M_2) model provided the best accuracy with an NMAE of 0.064. The VoD dataset proved more complex to model, as indicated by the higher NMAE values across all models compared to the KV dataset. Again, M_1 was the fastest, and M_3 was the slowest. All models were a clear improvement over the Naïve Baseline.

Table 2: Model Comparison for VoD Dataset (Target: DispFrames)

Model	**NMAE**	**Training Time (s)**
M_1 (Linear Regression)	0.119	0.009
M_2 (Random Forest)	0.064	2.006
M_3 (Neural Network)	0.115	627.180
Naïve Baseline	0.174	N/A

3 Neural Network Hyperparameter Search

A grid search was performed to find effective hyperparameters for the M_3 (Neural Network) models for both datasets.

KV Model (M_3) The KV model was constructed with two hidden layers. The best performance (NMAE 0.023) was achieved with the following hyperparameters:

- `neurons_layer1`: 64
- `neurons_layer2`: 4
- `learning_rate`: 0.008
- `dropout_rate`: 0.1

VoD Model (M_3) The VoD model used a deeper architecture with three hidden layers. The best performance (NMAE 0.115) was achieved with these parameters:

- `neurons_layer1`: 64
- `neurons_layer2`: 64
- `neurons_layer3`: 32

- `learning_rate`: 0.0007
- `dropout_rate`: 0.2

4 Target Value Distributions (Test Set)

The histogram and density plots for the target values on the test set (Step 6) revealed distinct distributions for the two services.

- **KV (ReadsAvg):** The distribution was observed to be unimodal and sharply peaked around its mean, indicating relatively low variance in the response time.
- **VoD (DispFrames):** The distribution was observed to be multimodal (primarily bimodal), with significant clusters of values. This suggests the system operates in different states, leading to different typical frame rates.

5 Time Series Analysis (M2 Model)

For the time series analysis (Step 5), the M_2 (Random Forest) model was selected as it provided the best NMAE for both datasets. The plots for both KV and VoD clearly showed the 'Estimation (M2)' line closely tracking the 'Measured' line. It successfully captured the dynamic peaks and troughs of the service metrics. In contrast, the 'Naive estimation' was a flat line representing the training set mean, which failed to capture any of the system's temporal behavior.

6 Estimation Error Analysis

The density plots of the estimation errors ($y_j - \hat{y}_j$) for all three models (Step 7) provided a clear comparison of model precision.

- For both KV and VoD datasets, the M_2 **(Random Forest)** model's error distribution was the narrowest and most sharply peaked around zero. This indicates the highest precision and lowest bias among the three models.
- The M_1 **(Linear Regression)** and M_3 **(Neural Network)** models showed wider and flatter error distributions, signifying larger and more varied errors in their predictions.

The error statistics for the VoD dataset were:

- **M1 Error:** Mean=0.097, Std=3.587, Min=-21.069, Max=7.952
- **M2 Error:** Mean=0.100, Std=2.654, Min=-17.409, Max=11.403
- **M3 Error:** Mean=-0.088, Std=4.720, Min=-25.868, Max=9.367

The M2 model had the lowest standard deviation of errors, reinforcing the observation from the density plot that it is the most consistent predictor.

7 Discussion and Conclusion

Based on the results from Task II, we can draw the following conclusions:

Accuracy: The M_2 (Random Forest Regression) model was the most accurate, achieving the lowest NMAE for both the simple KV dataset (0.019) and the more complex VoD dataset (0.064). All machine learning models significantly outperformed the Naïve Baseline, demonstrating their ability to learn the relationship between infrastructure measurements and service metrics.

Computational Overhead: The M_1 (Linear Regression) model was by far the fastest to train (0.015s and 0.009s). The M_2 (Random Forest) model was also very efficient, with training times of just 1.024s and 2.006s. The M_3 (Neural Network) model was extremely computationally expensive (335s and 627s), almost entirely due to the extensive hyperparameter search required to find a viable architecture.

Overall: For both services, the **Random Forest (M_2) model provides the best trade-off**, delivering the highest accuracy with a very low computational overhead. The Linear Regression (M_1) model, while fastest, is not as accurate, especially for the complex VoD data. The Neural Network (M_3) did not outperform the Random Forest in this instance and required significantly more time and effort to tune. This suggests that tree-based ensembles are highly effective for this type of tabular, high-dimensional operational data.