EP2420 Project 1

Task I: Data Exploration and Pre-processing

Estimating Service Metrics from Device Measurements

Author: Yiyi Miao

Date: October 31, 2025

Course: EP2420/EP272V

Datasets:

- `2025_JNSM_VoD_flashcrowd_2`
- `2025_JNSM_KV_flashcrowd_2`

Question 1: Design Matrix Description and Target Statistics

1.1 Design Matrix Dimensions

KV Dataset (X_0)

- **Number of sample rows:** 18,317
- **Number of feature columns:** 1,670

VoD Dataset (X_0)

- **Number of sample rows:** 18,317
- **Number of feature columns:** 1,670

1.2 Target Variable Statistics

KV Target (ReadsAvg) Statistics

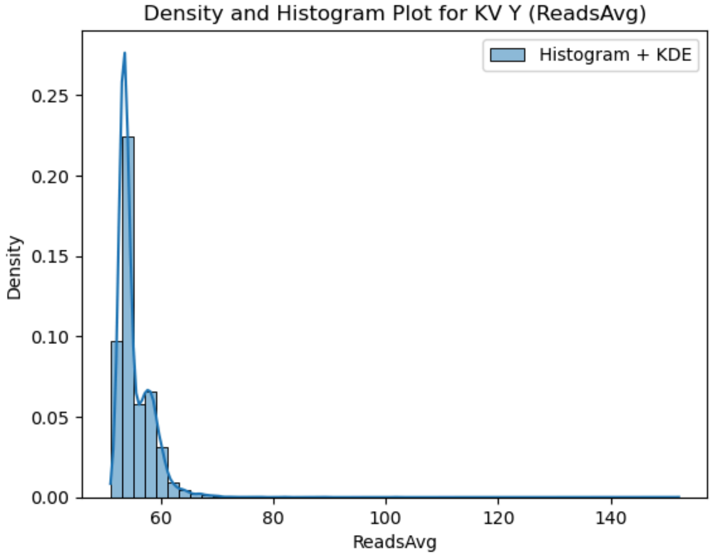
Statistic	Value
Mean	55.3 ms
Standard Deviation	3.13 ms
Maximum	152 ms
Minimum	51.1 ms
25th Percentile	53.3 ms
50th Percentile	54.1 ms
95th Percentile	60.5 ms

VoD Target (DispFrames) Statistics

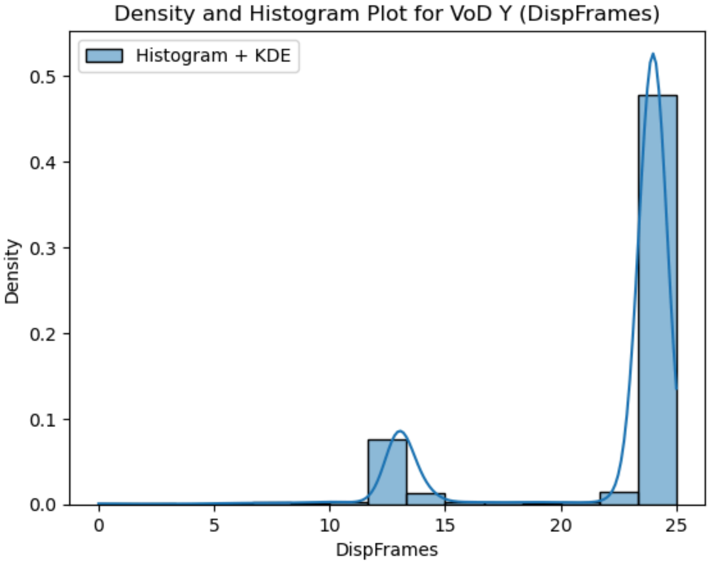
Statistic	Value
Mean	22.0 fps
Standard Deviation	4.33 fps
Maximum	25 fps
Minimum	0 fps
25th Percentile	24 fps
50th Percentile	24 fps
95th Percentile	24 fps

1.3 Target Distribution Visualization

KV Target Distribution



VoD Target Distribution



Observations:

- The KV target shows a concentrated distribution around 54-55 ms with a long right tail, indicating occasional high response times.
- The VoD target is heavily concentrated at 24 fps (standard video frame rate), with some samples dropping to lower values, suggesting service degradation events.

Question 2: Data Pre-processing Methods

Six different pre-processing methods were applied to both datasets, producing design matrices $X_1, X_2, \ldots, X_6$:

Pre-processing Transformations

Matrix	Method	Application Axis
X_1	L2 Normalization	Feature columns (axis=0)
X_2	L2 Normalization	Sample rows (axis=1)
X_3	Min-Max Scaling [0,1]	Feature columns
X_4	Min-Max Scaling [0,1]	Sample rows
X_5	Standardization (μ=0, σ=1)	Feature columns
X_6	Standardization (μ=0, σ=1)	Sample rows

Implementation Code

```
# KV Dataset Pre-processing
KV_X_1 = normalize(KV_X0, norm='l2', axis=0)
KV_X_2 = normalize(KV_X0, norm='l2', axis=1)
KV_X_3 = MinMaxScaler().fit_transform(KV_X0)
KV_X_4 = MinMaxScaler().fit_transform(KV_X0.T).T
KV_X_5 = StandardScaler().fit_transform(KV_X0)
KV_X_6 = StandardScaler().fit_transform(KV_X0.T).T

# VoD Dataset Pre-processing
VoD_X_1 = normalize(VoD_X0, norm='l2', axis=0)
VoD_X_2 = normalize(VoD_X0, norm='l2', axis=1)
VoD_X_3 = MinMaxScaler().fit_transform(VoD_X0)
VoD_X_4 = MinMaxScaler().fit_transform(VoD_X0.T).T
VoD_X_5 = StandardScaler().fit_transform(VoD_X0)
VoD_X_6 = StandardScaler().fit_transform(VoD_X0.T).T
```

Rationale:

- **Column-wise transformations** (X_1, X_3, X_5) normalize each feature independently, suitable when features have different scales.
- **Row-wise transformations** (X_2, X_4, X_6) normalize each sample, useful when the magnitude of observations varies significantly.

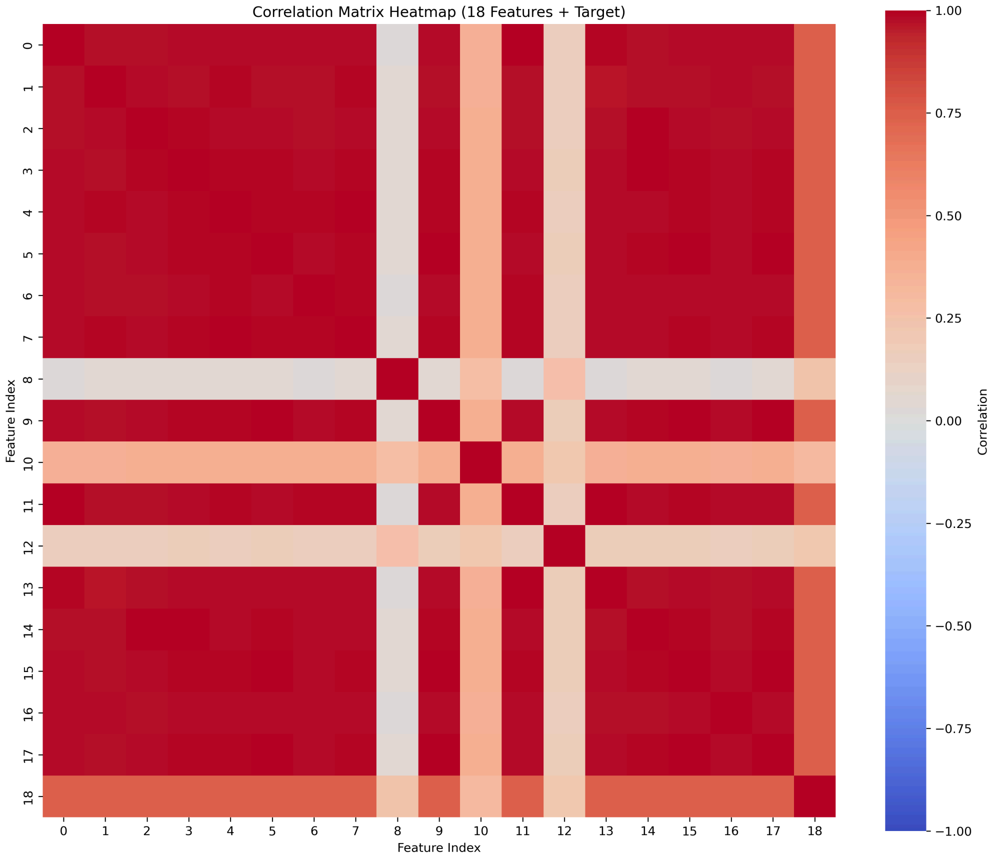
Question 3: Feature Selection and Correlation Analysis

3.1 Methodology

Using Random Forest Regressor with 50 trees, we selected the top 18 most important features from the original 1,670 features. The correlation matrix was computed for these 18 features plus the target variable, resulting in a 19×19 matrix.

3.2 Correlation Heatmap Analysis

KV Dataset Correlation Structure



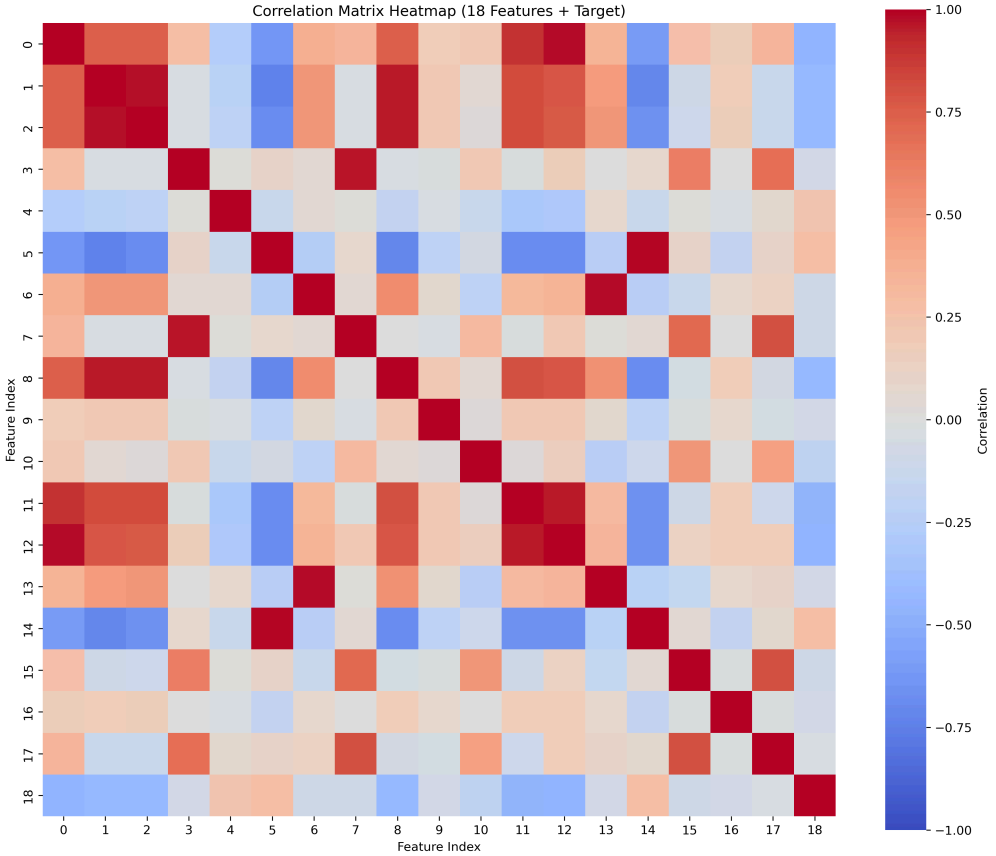
Key Observations:

1. **High Multicollinearity:** The heatmap is predominantly red, indicating strong positive correlations among most features. This suggests significant redundancy in the monitoring infrastructure.
2. **Feature Clusters:** Dense red blocks reveal groups of highly correlated features, likely measuring related system components (e.g., CPU metrics across multiple nodes).

3. **Target Correlations:** The bottom row/rightmost column shows that several features have strong positive correlations (>0.7) with ReadsAvg, indicating excellent predictive potential.
4. **Outlier Features:** A few lighter-colored stripes (around indices 8, 10, 12) indicate features with weaker correlations, possibly measuring independent system aspects.

Engineering Perspective: The Key-Value store's read performance is closely tied to specific infrastructure bottlenecks. The high multicollinearity suggests over-instrumentation, where multiple sensors capture redundant information about the same underlying system state.

VoD Dataset Correlation Structure



Key Observations:

1. **Complex Correlation Patterns:** Unlike KV, the VoD heatmap displays a diverse mixture of strong positive (dark red), strong negative (dark blue), and near-zero (white) correlations.
2. **Feature Independence:** More white and blue regions indicate greater independence between features, suggesting the video streaming infrastructure has more diverse and decoupled monitoring points.
3. **Mixed Target Relationships:** Correlations with DispFrames vary widely—some features show positive relationships, others negative, and some near-zero. This indicates that video quality depends on multiple, sometimes opposing factors.
4. **Block Structure:** Visible block patterns suggest subsystems (encoding, networking, storage) with internal correlations but limited cross-system correlation.

Engineering Perspective: Video frame rate is influenced by complex, multi-factor interactions. The presence of both positive and negative correlations suggests trade-offs in the system (e.g., higher encoding quality may reduce frame rate, or bandwidth allocation may have inverse relationships). This complexity implies that non-linear models may be necessary for accurate prediction.

Comparative Insight: The stark difference between KV (homogeneous, highly correlated) and VoD (heterogeneous, mixed correlations) reflects the fundamental architectural differences: KV operations are primarily compute-bound with predictable dependencies, while video streaming involves diverse resources with complex interdependencies.

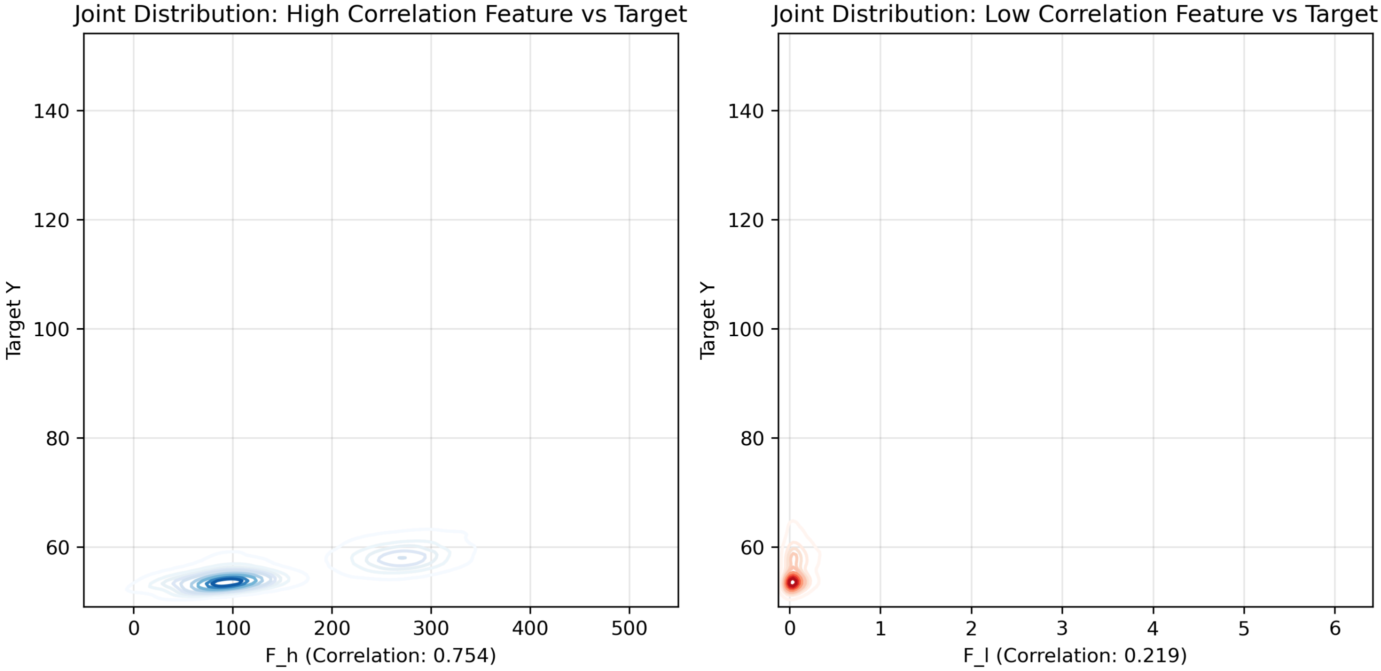
Question 4: Joint Distribution Analysis

4.1 Feature Selection for Comparison

For each dataset, we identified:

- **F_h:** Feature with highest absolute correlation to target (best predictor)
- **F_l:** Feature with lowest absolute correlation to target (worst predictor)

4.2 KV Dataset Joint Distributions



High Correlation Feature (F_h, Correlation: 0.754)

Visual Pattern: The plot reveals a clear, strong positive relationship between F_h and the target. The concentration of density (dark blue regions) forms a diagonal pattern ascending from lower-left to upper-right.

Interpretation:

- As F_h increases, ReadsAvg (response time) consistently increases
- The tight clustering along the trend line indicates low variance and high predictability

- This feature likely represents a direct performance bottleneck (e.g., CPU utilization, queue length, or concurrent request count)

Engineering Insight: This metric captures a resource that directly constrains read operations. The linear relationship suggests proportional scaling—doubling F_h roughly doubles response time.

Low Correlation Feature (F_l, Correlation: 0.219)

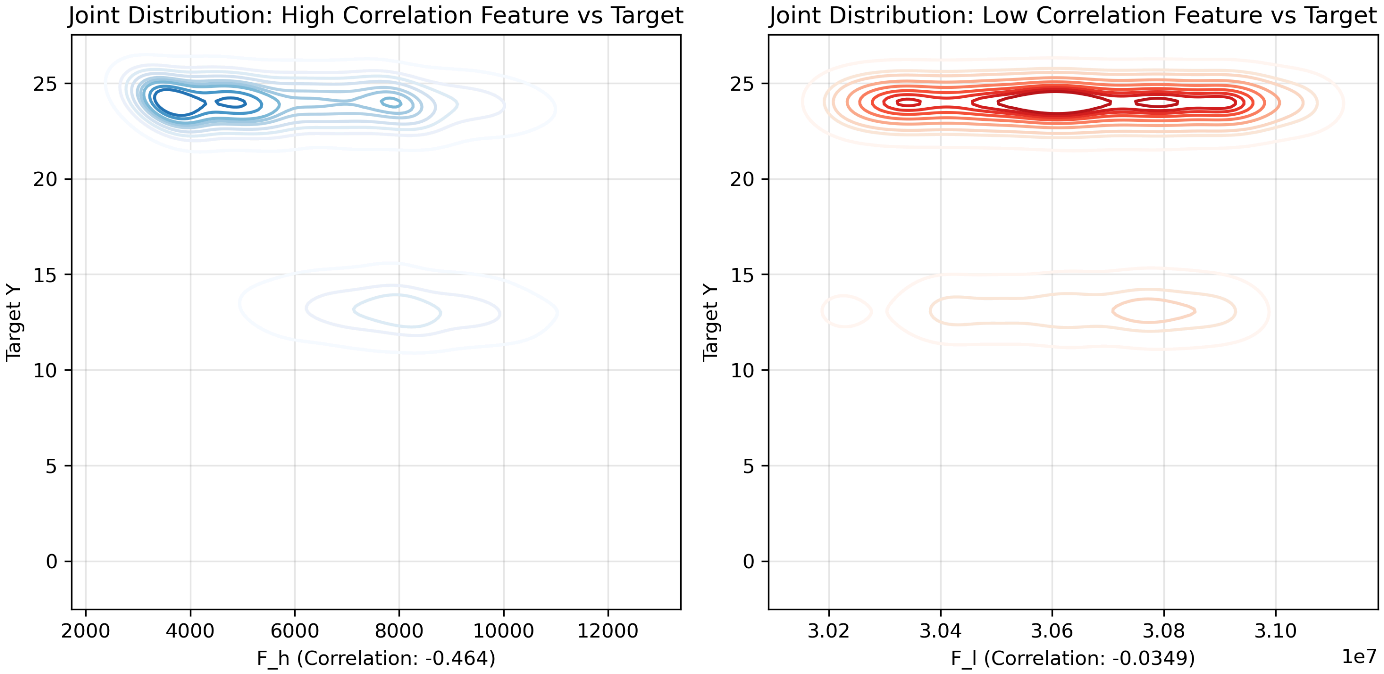
Visual Pattern: The plot shows a concentrated vertical blob with no horizontal trend. The data is tightly clustered at low F_l values regardless of target values.

Interpretation:

- Changes in F_l provide no information about ReadsAvg
- The target varies across its full range while F_l remains nearly constant
- This feature has minimal variance and no predictive power

Engineering Insight: This metric likely monitors a non-critical or over-provisioned resource (e.g., disk I/O on a cache-serving system, or memory on a lightly-loaded node). Its lack of variation suggests it never becomes a bottleneck.

4.3 VoD Dataset Joint Distributions



High Correlation Feature (F_h, Correlation: -0.464)

Visual Pattern: The plot exhibits a clear negative relationship with two distinct density clusters connected by a diagonal band sloping downward from left to right.

Interpretation:

- When F_h is low (~4,000), DispFrames clusters near maximum (25 fps)
- When F_h is high (~9,000), DispFrames drops to minimum (~13 fps)
- The inverse relationship suggests F_h represents a cost or load metric

Engineering Insight: This feature likely measures resource consumption or system load (e.g., CPU usage, encoding complexity, or network congestion). As this metric increases, the system cannot maintain high frame rates, resulting in degraded video quality. The bimodal distribution suggests the system operates in two regimes: normal (high frame rate, low load) and degraded (low frame rate, high load).

Low Correlation Feature (F_l, Correlation: -0.0349)

Visual Pattern: The plot shows two horizontal bands with no relationship to the x-axis. The density is split between high frame rates (~24 fps) and low frame rates (~13 fps), independent of F_l.

Interpretation:

- F_l values span their range regardless of video quality
- The horizontal banding reflects the bimodal nature of DispFrames
- No predictive relationship exists between F_l and target

Engineering Insight: This metric monitors a system component that is not involved in the video delivery pipeline's critical path (e.g., storage I/O on a fully-cached system, or memory usage in a non-bottleneck service). Its independence from frame rate confirms it can be excluded from prediction models.

4.4 Comparative Summary

The joint distribution analysis validates the correlation-based feature selection:

Aspect	High Correlation (F_h)	Low Correlation (F_l)
Pattern	Clear directional trend	Random scatter or independence
Predictive Value	High—can estimate target	None—provides no information
Engineering Role	Critical bottleneck resource	Non-critical or over-provisioned
Model Utility	Essential feature	Can be safely removed

From an engineering perspective, F_h represents key performance indicators directly impacting service quality, while F_l corresponds to secondary metrics with negligible influence on user experience.

End of Task I Report