Workload Characterization Techniques

Raj Jain
Washington University in Saint Louis
Saint Louis, MO 63130
Jain@cse.wustl.edu

These slides are available on-line at:

http://www.cse.wustl.edu/~jain/cse567-08/

- Terminology
- Components and Parameter Selection
- Workload Characterization Techniques: Averaging, Single Parameter Histograms, Multi-parameter Histograms, Principal Component Analysis, Markov Models, Clustering
- Clustering Method: Minimum Spanning Tree, Nearest Centroid
- Problems with Clustering

Terminology

- User = Entity that makes the service request
- Workload components:
  - Applications
  - Sites
  - User Sessions
- Workload parameters or Workload features: Measured quantities, service requests, or resource demands. For example: transaction types, instructions, packet sizes, source-destinations of a packet, and page reference pattern.

Components and Parameter Selection

- The workload component should be at the SUT interface.
- Each component should represent as homogeneous a group as possible. Combining very different users into a site workload may not be meaningful.
- Domain of the control affects the component: Example: mail system designer are more interested in determining a typical mail session than a typical user session.
- Do not use parameters that depend upon the system, e.g., the elapsed time, CPU time.

Components (Cont)

- Characteristics of service requests:
  - Arrival Time
  - Type of request or the resource demanded
  - Duration of the request
  - Quantity of the resource demanded, for example, pages of memory
- Exclude those parameters that have little impact.

Workload Characterization Techniques

1. Averaging
2. Single-Parameter Histograms
3. Multi-parameter Histograms
4. Principal Component Analysis
5. Markov Models
6. Clustering

Averaging

- Mean $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$
- Standard deviation: $s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$
- Coefficient Of Variation: $s/\bar{x}$
- Mode (for categorical variables): Most frequent value
- Median: 50-percentile

Case Study: Program Usage in Educational Environments

Data	Average		Coef. of Variation
CPU time (VAX-11/780™)	2.19	seconds	40.23
Elapsed time	73.90	seconds	8.59
Number of direct writes	8.20		53.59
Direct write bytes	10.21	kilobytes	82.41
Size of direct writes	1.25	kilobytes	
Number of direct reads	22.64		25.65
Direct read bytes	49.70	kilobytes	21.01
Size of direct reads	2.20	kilobytes	
Number of buffered writes	52.84		11.80
Buffered write bytes	978.04	bytes	9.98
Size of buffered writes	18.51	bytes	

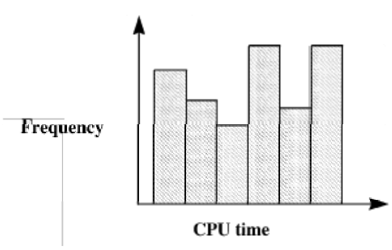
- High Coefficient of Variation

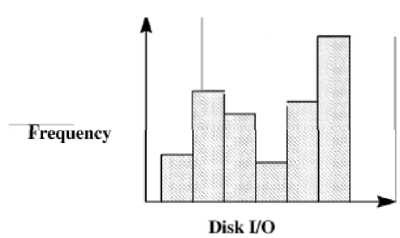
Characteristics of an Average Editing Session

Data	Average		Coef. of Variation
CPU time (VAX-11/780)	2.57	seconds	3.54
Elapsed time	265.45	seconds	2.34
Number of direct writes	19.74		4.33
Direct write bytes	13.46	Kilo-bytes	3.87
Size of direct writes	0.68	kilo-bytes	
Number of direct reads	37.77		3.73
Direct read bytes	36.93	Kilo-bytes	3.16
Size of direct reads	0.98	kilo-bytes	
Number of buffered writes	199.06		4.30
Buffered write bytes	3314.95 bytes	3.04	
Size of buffered writes	16.65	bytes	

- Reasonable variation

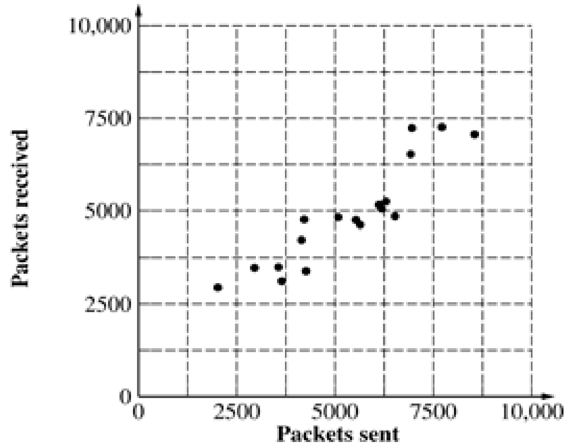
Single Parameter Histograms





- *n* buckets × *m* parameters × *k* components values.
- Use only if the variance is high.
- Ignores correlation among parameters.

Multi-parameter Histograms



- Difficult to plot joint histograms for more than two parameters.

Principal Component Analysis

- **Key Idea**: Use a weighted sum of parameters to classify the components.
- Let x_{ij} denote the ***i***th parameter for jth component.

$$y_j = \sum_{i=1}^{n} w_i \, x_{ij}$$

- Principal component analysis assigns weights w_i's such that y_j's provide the maximum discrimination among the components.
- The quantity y_j is called the principal factor.
- The factors are ordered. First factor explains the highest percentage of the variance.

Principal Component Analysis (Cont)

- Statistically:
  - The y's are linear combinations of x's:

$$y_i = \sum_{j=1}^{n} a_{ij} x_j$$

Here, a_{ij} is called the loading of variable x_j on factor y_i.

  - The y's form an orthogonal set, that is, their inner product is zero:

$$\langle y_i, y_j \rangle = \sum_k a_{ik} a_{kj} = 0$$

This is equivalent to stating that y_i's are uncorrelated to each other.

  - The y's form an ordered set such that y_1 explains the highest percentage of the variance in resource demands.

Finding Principal Factors

- Find the correlation matrix.
- Find the eigen values of the matrix and sort them in the order of decreasing magnitude.
- Find corresponding eigen vectors.
  These give the required loadings.

Principal Component Example

Obs. No.	Variables x_s	x_r	Normalized Variables x'_s	x'_r	Principal Factors y_1	y_2
1	7718	7258	1.359	1.717	2.175	-0.253
2	6958	7232	0.922	1.698	1.853	-0.549
3	8551	7062	1.837	1.575	2.413	0.186
4	6924	6526	0.903	1.186	1.477	-0.200
5	6298	5251	0.543	0.262	0.570	0.199
6	6120	5158	0.441	0.195	0.450	0.174
7	6184	5051	0.478	0.117	0.421	0.255
8	6527	4850	0.675	-0.029	0.457	0.497
9	5081	4825	-0.156	-0.047	-0.143	-0.077
10	4216	4762	-0.652	-0.092	-0.527	-0.396
17	3644	3120	-0.981	-1.283	-1.601	0.213
18	2020	2946	-1.914	-1.409	-2.349	-0.357
$\sum x$	96336	88009	0.000	0.000	0.000	0.000
$\sum x^2$	567119488	462661024	17.000	17.000	32.565	1.435
Mean	5352.0	4889.4	0.000	0.000	0.000	0.000
Std. Dev.	1741.0	1379.5	1.000	1.000	1.384	0.290

Principal Component Example

- Compute the mean and standard deviations of the variables:

$$\bar{x}_s = \frac{1}{n}\sum_{i=1}^{n} x_{si} = \frac{96336}{18} = 5352.0$$

$$\bar{x}_r = \frac{1}{n}\sum_{i=1}^{n} x_{ri} = \frac{88009}{18} = 4889.4$$

Principal Component (Cont)

$$s_{x_s}^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_{si}-\bar{x}_s)^2$$
$$= \frac{1}{n-1}\left[\left(\sum_{i=1}^{n} x_{si}^2\right) - n * \bar{x}_s^2\right]$$
$$= \frac{567119488 - 18 \times 5353^2}{17} = 1741.0^2$$

- Similarly:

$$s_{x_r}^2 = \frac{462661024 - 18 \times 4889.4^2}{17} = 1379.5^2$$

Principal Component (Cont)

- Normalize the variables to zero mean and unit standard deviation. The normalized values x_s' and x_r' are given by:

$$x_s' = \frac{x_s - \bar{x}_s}{s_{x_s}} = \frac{x_s - 5352}{1741}$$
$$x_r' = \frac{x_r - \bar{x}_r}{s_{x_r}} = \frac{x_r - 4889}{1380}$$

Principal Component (Cont)

- Compute the correlation among the variables:

$$R_{x_s,x_r} = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_{si}-\bar{x}_s)(x_{ri}-\bar{x}_r)}{s_{x_s}s_{x_r}} = 0.916$$

- Prepare the correlation matrix:

$$\mathbf{C} = \begin{bmatrix} 1.000 & 0.916 \\ 0.916 & 1.000 \end{bmatrix}$$

Principal Component (Cont)

- Compute the eigen values of the correlation matrix: By solving the characteristic equation:

$$|\lambda I - C| = \begin{vmatrix} \lambda - 1 & -0.916 \\ -0.916 & \lambda - 1 \end{vmatrix} = 0$$

$$(\lambda - 1)^2 - 0.916^2 = 0$$

- The eigen values are 1.916 and 0.084.

Principal Component (Cont)

- Compute the eigen vectors of the correlation matrix. The eigen vectors q_1 corresponding to λ_1=1.916 are defined by the following relationship:

  $$\{C\}\{q\}_1 = \lambda_1 \{q\}_1$$

  or:

  $$\begin{bmatrix} 1.000 & 0.916 \\ 0.916 & 1.000 \end{bmatrix} \times \begin{bmatrix} q_{11} \\ q_{21} \end{bmatrix} = 1.916 \begin{bmatrix} q_{11} \\ q_{21} \end{bmatrix}$$

  or:

  $$q_{11}=q_{21}$$

Principal Component (Cont)

- Restricting the length of the eigen vectors to one:

  $$\mathbf{q}_1 = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} \quad \mathbf{q}_2 = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix}$$

- Obtain principal factors by multiplying the eigen vectors by the normalized vectors:

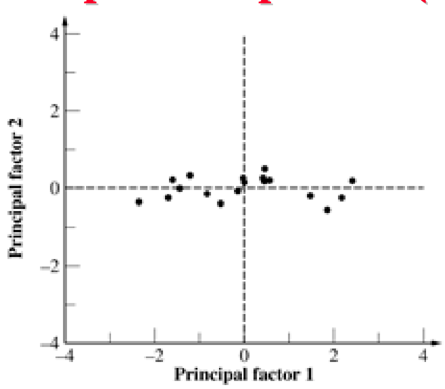
  $$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \frac{x_s - 5352}{1741} \\ \frac{x_r - 4889}{1380} \end{bmatrix}$$

- Compute the values of the principal factors.
- Compute the sum and sum of squares of the principal factors.

Principal Component (Cont)

- The sum must be zero.
- The sum of squares give the percentage of variation explained.

Principal Component (Cont)

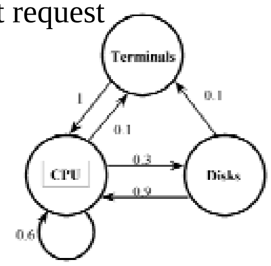


- The first factor explains 32.565/(32.565+1.435) or 95.7% of the variation.
- The second factor explains only 4.3% of the variation and can, thus, be ignored.

Markov Models

- Markov
  ⇒ the next request depends only on the last request
- Described by a transition matrix:

From/To	CPU	Disk	Terminal
CPU	0.6	0.3	0.1
Disk	0.9	0	0.1
Terminal	1	0	0



- Transition matrices can be used also for application transitions. E.g., P(Link|Compile)
- Used to specify page-reference locality.
  P(Reference module *i* | Referenced module *j*)

Transition Probability

- Given the same relative frequency of requests of different types, it is possible to realize the frequency with several different transition matrices.
- If order is important, measure the transition probabilities directly on the real system.
- Example: Two packet sizes: Small (80%), Large (20%)
  - An average of four small packets are followed by an average of one big packet, e.g., ssssbssssbssss.

Current	Next packet	
Packet	Small	Large
Small	0.75	0.25
Large	1	0

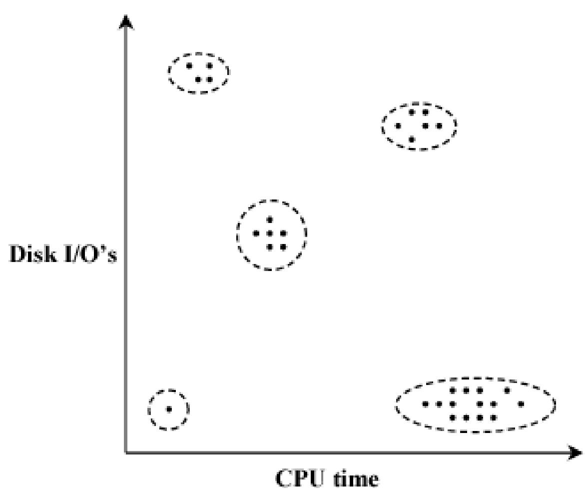
Transition Probability (Cont)

- Eight small packets followed by two big packets.

Current Packet	Next packet Small	Next packet Large
Small	0.875	0.125
Large	0.5	0.5

- Generate a random number *x*.
  $x \leq 0.8 \Rightarrow$ generate a small packet;
  otherwise generate a large packet.

Clustering



Clustering Steps

1. Take a sample, that is, a subset of workload components.
2. Select workload parameters.
3. Select a distance measure.
4. Remove outliers.
5. Scale all observations.
6. Perform clustering.
7. Interpret results.
8. Change parameters, or number of clusters, and repeat steps 3-7.
9. Select representative components from each cluster.

1. Sampling

- In one study, 2% of the population was chosen for analysis; later 99% of the population could be assigned to the clusters obtained.
- Random selection
- Select top consumers of a resource.

2. Parameter Selection

- Criteria:
  - Impact on performance
  - Variance
- Method: Redo clustering with one less parameter
- Principal component analysis: Identify parameters with the highest variance.

3. Transformation

- If the distribution is highly skewed, consider a function of the parameter, e.g., log of CPU time

4. Outliers

- Outliers = data points with extreme parameter values
- Affect normalization
- Can exclude only if that do not consume a significant portion of the system resources. Example, backup.

5. Data Scaling

1. Normalize to Zero Mean and Unit Variance:

$$x'_{ik} = \frac{x_{ik} - \bar{x}_k}{s_k}$$

2. Weights:

$x_{ik}' = w_k x_{ik}$

$w_k \propto$ relative importance or $w_k = 1/s_k$

3. Range Normalization:

$$x'_{ik} = \frac{x_{ik} - x_{min,k}}{x_{max,k} - x_{min,k}}$$

Affected by outliers.

Data Scaling (Cont)

- Percentile Normalization:

$$x'_{ik} = \frac{x_{ik} - x_{2.5,k}}{x_{97.5,k} - x_{2.5,k}}$$

Distance Metric

1. Euclidean Distance: Given $\{x_{i1}, x_{i2}, ..., x_{in}\}$ and $\{x_{j1}, x_{j2}, ..., x_{jn}\}$

   $$d=\{\sum_{k=1}^{n} (x_{ik}-x_{jk})^2\}^{0.5}$$

2. Weighted-Euclidean Distance:

   $$d=\sum_{k=1}^{n} \{a_k(x_{ik}-x_{jk})^2\}^{0.5}$$

   Here a_k, k=1,2,…,n are suitably chosen weights for the n parameters.

3. Chi-Square Distance:

   $$d = \sum_{k=1}^{n} \left\{ \frac{(x_{ik} - x_{jk})^2}{x_{ik}} \right\}$$

Distance Metric (Cont)

- The Euclidean distance is the most commonly used distance metric.
- The weighted Euclidean is used if the parameters have not been scaled or if the parameters have significantly different levels of importance.
- Use Chi-Square distance only if $x_{.k}$'s are close to each other. Parameters with low values of $x_{.k}$ get higher weights.

Clustering Techniques

- Goal: Partition into groups so the members of a group are as similar as possible and different groups are as dissimilar as possible.
- Statistically, the intragroup variance should be as small as possible, and inter-group variance should be as large as possible.

  Total Variance = Intra-group Variance + Inter-group Variance

Clustering Techniques (Cont)

- **Nonhierarchical techniques**: Start with an arbitrary set of k clusters, Move members until the intra-group variance is minimum.
- **Hierarchical Techniques**:
  - Agglomerative: Start with n clusters and merge
  - Divisive: Start with one cluster and divide.
- Two popular techniques:
  - Minimum spanning tree method (agglomerative)
  - Centroid method (Divisive)

Minimum Spanning Tree-Clustering Method

1. Start with $k = n$ clusters.
2. Find the centroid of the i^{th} cluster, i=1, 2, …, k.
3. Compute the inter-cluster distance matrix.
4. Merge the the nearest clusters.
5. Repeat steps 2 through 4 until all components are part of one cluster.

Minimum Spanning Tree Example

Program	CPU Time	Disk I/O
A	2	4
B	3	5
C	1	6
D	4	3
E	5	2

- Step 1: Consider five clusters with ith cluster consisting solely of ith program.
- Step 2: The centroids are {2, 4}, {3, 5}, {1, 6}, {4, 3}, and {5, 2}.

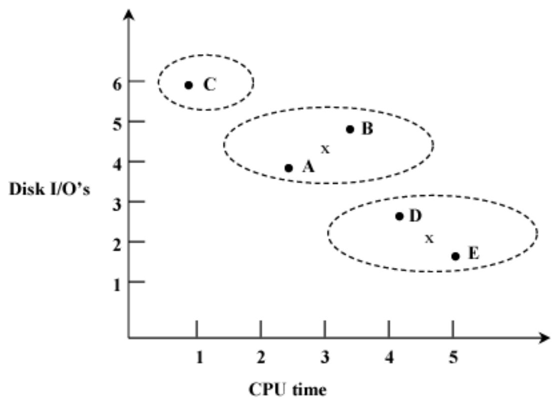
Spanning Tree Example (Cont)

- Step 3: The Euclidean distance is:

	Program				
Program	A	B	C	D	E
A	0	$\sqrt{2}$	$\sqrt{5}$	$\sqrt{5}$	$\sqrt{13}$
B		0	$\sqrt{5}$	$\sqrt{5}$	$\sqrt{13}$
C			0	$\sqrt{18}$	$\sqrt{32}$
D				0	$\sqrt{2}$
E					0

- Step 4: Minimum inter-cluster distance = $\sqrt{2}$. Merge A+B, D+E.

Spanning Tree Example (Cont)



- Step 2: The centroid of cluster pair AB is {(2+3)÷ 2, (4+5) ÷ 2}, that is, {2.5, 4.5}. Similarly, the centroid of pair DE is {4.5, 2.5}.

Spanning Tree Example (Cont)

- Step 3: The distance matrix is:

	Program		
Program	AB	C	DE
AB	0	$\sqrt{4.5}$	$\sqrt{10.25}$
C		0	$\sqrt{24.4}$
DE			0

- Step 4: Merge AB and C.
- Step 2: The centroid of cluster ABC is {(2+3+1) ÷ 3, (4+5+6) ÷ 3}, that is, {2, 5}.

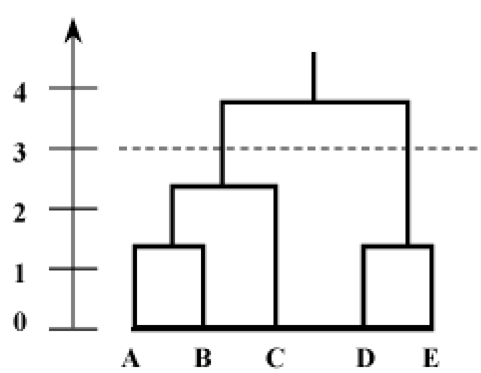
Spanning Tree Example (Cont)

- Step 3: The distance matrix is:

	Program	
Program	ABC	DE
ABC	0	$\sqrt{12.5}$
DE		0

- Step 4: Minimum distance is 12.5.
  Merge ABC and DE $\Rightarrow$ Single Custer ABCDE

Dendogram



- Dendogram = Spanning Tree
- Purpose: Obtain clusters for any given maximum allowable intra-cluster distance.

Nearest Centroid Method

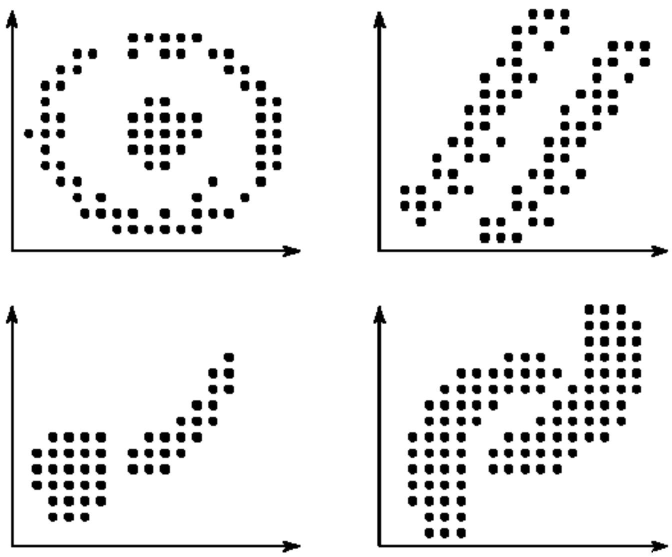
- Start with $k = 1$.
- Find the centroid and intra-cluster variance for i^{th} cluster, $i = 1, 2, \ldots, k$.
- Find the cluster with the highest variance and arbitrarily divide it into two clusters.
  - Find the two components that are farthest apart, assign other components according to their distance from these points.
  - Place all components below the centroid in one cluster and all components above this hyper plane in the other.
- Adjust the points in the two new clusters until the inter-cluster distance between the two clusters is maximum.
- Set $k = k+1$. Repeat steps 2 through 4 until $k = n$.

Cluster Interpretation

- Assign all measured components to the clusters.
- Clusters with very small populations and small total resource demands can be discarded.

  (Don't just discard a small cluster)
- Interpret clusters in functional terms, e.g., a business application, Or label clusters by their resource demands, for example, CPU-bound, I/O-bound, and so forth.
- Select one or more representative components from each cluster for use as test workload.

Problems with Clustering



Problems with Clustering (Cont)

- Goal: Minimize variance.
- The results of clustering are highly variable. No rules for:
  - Selection of parameters
  - Distance measure
  - Scaling
- Labeling each cluster by functionality is difficult.
  - In one study, editing programs appeared in 23 different clusters.
- Requires many repetitions of the analysis.

Summary

- Workload Characterization = Models of workloads
- Averaging, Single parameter histogram, multi-parameter histograms, …
- Principal component analysis consists of finding parameter combinations that explain the most variation
- Clustering: divide workloads in groups that can be represented by a single benchmark

Exercise 6.1

- The CPU time and disk I/Os of seven programs are shown in Table below. Determine the equation for principal factors.

Table 1: Data for Principal Component Exercise

Program Name	Function	CPU Time	Number of I/Os
TKB	Linker	14	2735
MAC	Assembler	13	253
COBOL	Compiler	8	27
BASIC	Compiler	6	27
PASCAL	Compiler	6	12
EDT	Text Editor	4	91
SOS	Text Editor	1	33

Exercise 6.2

- Using a spanning-tree algorithm for cluster analysis, prepare a Dendogram for the data shown in Table below. Interpret the result of your analysis.

Table 1: Data for Principal Component Exercise

Program Name	Function	CPU Time	Number of I/Os
TKB	Linker	14	2735
MAC	Assembler	13	253
COBOL	Compiler	8	27
BASIC	Compiler	6	27
PASCAL	Compiler	6	12
EDT	Text Editor	4	91
SOS	Text Editor	1	33

Homework

- Read chapter 6
- Submit answers to exercises 6.1 and 6.2