

Yazılım Tasarımı Kalitesi

Lisans: <https://creativecommons.org/licenses/by-nc-nd/4.0/deed.tr>

Ölçme Bilimi "Metroloji (Metrology)" ile İlgili Temel Bilgiler

Nasıl Ölçeceğiz?

"Ölçemediğiniz şeyi kontrol edemezsiniz."
Tom DeMarco (1940-), ABD Yazılım Uzmanı

T. DeMarco, Controlling software projects : management, measurement & estimation. New York NY: Prentice Hall/Yourdon Press, 1982.

Okunacak yazılar:

- N. Fenton, "Software measurement: a necessary scientific basis," IEEE Transactions on Software Engineering, vol. 20, no. 3, pp. 199–206, Mar. 1994. <http://dx.doi.org/10.1109/32.268921>
- B. Kitchenham, S. L. Pfleeger, and N. Fenton, "Towards a framework for software measurement validation," IEEE Transactions on Software Engineering, vol. 21, no. 12, pp. 929–944, Dec. 1995. <http://dx.doi.org/10.1109/32.489070>

Kaynak Kitap:

- Fenton, N and Bieman, J., "Software metrics: A rigorous and practical approach", 3/e, CRC Press, 2015

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.1

Yazılım Tasarımı Kalitesi

Bilgi Engeli (Intelligence Barrier)

Gerçek dünyadaki bazı varlıklar (özellikle yazılım varlıkları) hakkında gözlem yoluyla değerlendirme yapmak kolay değildir (hatta bazen mümkün değildir). Örneğin, aynı işi yapan iki yazılımdan hangisinin daha karmaşık olduğunu sadece gözlemle belirleyebilmek için deneyimli yazılım uzmanlarının bu yazılımları incelemesi ve yorumlaması gerekir.

Ayrıca, bir sınıf gelecekte hata çıkarır mı, bunu öngörmek de kolay olmaz. Bu tür karşılaştırmalar ve yorumlar ölçme yoluyla dolaylı olarak yapılır. Değerlendirme gerçek dünyada değil, ölçme ile elde edilen sayısal değerler üzerinde yapılır.

Sahip olduğumuz bilgiler (Gerçek Dünya) → Ölçme → Sayısal değerler, ilişkiler, işlemler (Matematiksel Dünya)

Bilgi Engeli → Değerlendirme, Yorumlama → Sayısal sonuçlar (Matematiksel işlemler, İstatistik)

Hedef → Gerçek dünya ile ilgili sonuçlar

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.2

Yazılım Tasarımı Kalitesi

Ölçmenin İki Temel Kullanım Şekli:

1. Değerlendirme (Assessment, judgment):

Yazılımın/projenin o andaki durumu değerlendirilir ve kararlar verilir.

Örnekler:

Yazılım modülleri olgunlaşmış mıdır, hata yoğunlukları belli bir düzeyin altında mıdır, yoksa bu modüller iyileştirilmeli midir?

İki yazılım modülü anlaşılabilirlik yönünden karşılaştırılıp uygun olan seçilebilir.

Bir modülün eski ve yeni sürümlerdeki karmaşıklığı karşılaştırılır, iyileşme var mı belirlenir.

Kestirim (estimation): Sistemin doğrudan ölçülemeyen niteliklerinin o andaki durumu ile ilgili tahmin (kestirim) ölçülebilen nitelikler cinsinden oluşturulur.

Örnek:

Programda henüz bulunmamış hata sayısı, önceki hataların sayısı, sınıf sayısı, metod sayısı gibi ölçüler kullanılarak kestirilebilir.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.3

Yazılım Tasarımı Kalitesi

Ölçmenin İki Temel Kullanım Şekli (devamı):

2. Öngörü (Prediction):

Yazılımın/projenin gelecekteki durumu hakkında değerlendirme yapılır.

a) Bir karakteristik ile ilgili geçmişte toplanan verilerden (ölçümlerden) bir model oluşturularak aynı karakteristik için gelecekteki değeri öngörülür. Burada makine öğrenmesi yöntemleri kullanılır. Örneğin geçmişte sık hata çıkaran sınıflara benzer özelliklere sahip sınıfların gelecekte sık hata çıkarması beklenebilir.

b) Gelecekteki değeri öngörülmesi istenen karakteristik, o anda ölçülebilen başka niteliklerin ölçülmesi cinsinden ifade edilir (korelasyon). Örneğin, bakım sırasındaki güncelleme maliyeti o andaki karmaşıklık ölçülmesi cinsinden öngörülebilir.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.4

Yazılım Tasarımı Kalitesi

Ölçme Bilimi "Metroloji (Metrology)" ile İlgili Temel Bilgiler

Nasıl Ölçeceğiz?

Ölçme (Measurement):

Ölçme, gerçek dünyadaki fiziksel ya da sezgisel (empirical) varlıkların (entity) bazı niteliklerine (attribute) belli kurallara uygun şekilde sayılar ya da semboller karşılığında dönüştürmektir.

Varlık (entity): Ölçmeye konu olan fiziksel ya da sezgisel varlıklardır. Örneğin; kişi, sınıf, nesne, metod, test süreci, bilgisayar donanımı. Hatırlatma: Yazılım dünyasında varlıklar üç alandan kaynaklanır: Yazılım ürünleri, yazılım süreçleri, kaynaklar.

Nitelik (attribute): Varlığın ölçülmek istenen özelliği. Örneğin; kişinin boyu, sınıfın işlevsel uyumu, nesnenin bellekte kapladığı yer, metodun uzunluğu. Nitelikler çok boyutlu (multidimensional) olabilir. Örneğin hız (velocity) hem birim zamandaki yer değiştirme (speed) hem de yön bilgisi içerebilir.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.5

Yazılım Tasarımı Kalitesi

Ölçme: A fiziksel ya da sezgisel gerçek dünya varlıkları kümesi, B formel (sayısal, matematiksel) varlıklar kümesi (örneğin reel sayılar) olmak üzere, m ölçmesi iki küme arasında bire bir karşı düşürme (mapping) olarak tanımlanır.

m: A → B

A **m** **B**

Gerçek (sezgisel) dünya (Empirical world) Matematiksel (formel) dünya (Mathematical, formal world)

m ölçme işlemi bire bir (one-to-one) bir ilişki olduğu için A'daki her varlığın B'de mutlaka bir ve sadece bir karşılığı (ölçme değeri) vardır.

Örneğin $B = \mathbb{R}^+$ olabilir.
 $\mathbb{R}^+$: Pozitif reel sayılar kümesi

Değer (value): Ölçme işleminde niteliklere çoğunlukla sayısal değerler karşılığında düşürülür. Başka değerler de kullanılabilir. Örneğin, yazılımdaki hatalar şu şekilde değerlendirilebilir:

- Gereksinim hatası, tasarım hatası, kod hatası veya
- Çok kritik, orta kritik, az kritik.

Elbette bu değerler sayılarla da ifade edilebilir.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.6

Yazılım Tasarımı Kalitesi Lisans: <https://creativecommons.org/licenses/by-nc-nd/4.0/deed.tr>

Ölçmede Temsil Teorisi (Representational Theory of Measurement)

Ölçmenin amacı, gerçek dünyadaki varlıklar hakkında onları değerlendirebilecek (*assessment*) ve gelecekteki durumları hakkında öngörüler (*prediction*) yapabilecek bilgiler elde etmektir (bilgi engelini aşmak, bkz. 3.2).

Ölçme yönteminde **temsil (representation) özelliği** olmalıdır.

Gerçek dünyada iki varlık arasında belli bir niteliğe göre var olduğu bilinen (öyle olduğu algılanan) bağıntı (ilişki) (*relation*), ölçme sonucu elde edilen değerler arasında da bulunmalıdır.

Örneğin, sezgisel (gözlemsel) olarak k1 kişinin boyunun k2 kişisinden "daha uzun" olduğu gözlemleniyorsa, ölçme sonucu k1 kişinin boyunun k2 kişisinden daha uzun olduğunu göstermelidir.

Temsil özelliği, matematiksel dünyadan gerçek dünyaya doğru da geçerli olmalıdır.

Örneğin, yazılım sınıflarının anlaşılabilirliğini ölçebilen bir yöntem A sınıfına B sınıfından daha yüksek bir değer atadıysa gerçek yazılımda A sınıfı B sınıfından daha kolay anlaşılır olmalıdır.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.7

Yazılım Tasarımı Kalitesi

Ölçme ve Bağıntılar (Measurement and Relationships)

Temsil koşulunu (*representation condition*) açıklamak için önce **bağıntı sistemi** tanımlanacaktır.

Bağıntı sistemi (relational system): Tanım üç küme içerir: (S,R,O).

Tanım: (S, $r_1, \dots, r_n, op_1, \dots, op_m$) Belli bir niteliğe (*attribute*) göre tanımlanır.

S: İncelenen niteliğe (*attribute*) sahip tüm varlıklar kümesi,

$R = \{r_1, \dots, r_n\}$: Varlıkların niteliği üzerinde tanımlı bağıntılar (ilişkiler) (*relation*),

$O = \{op_1, \dots, op_m\}$: Varlıklar üzerinde tanımlı ikili işlemler (*operation*),

Örnek: Gerçek dünyadaki (sezgisel) bir bağıntı sistemi

İlgilenilen nitelik: uzunluk.

S: Elimizdeki tüm ipler,

$R = \{r_1, \dots, r_n\}$: ip1, ip2'den "daha uzun". İp1 ile ip2'nin uzunlukları "aynı".

$O = \{op_1, \dots, op_m\}$: İp1 ve ip2 "eklenip" ip3 elde ediliyor.

Örnek: Sayısal (formel) bir bağıntı sistemi

S: Reel sayılar,

$R = \{r_1, \dots, r_n\}$: büyük, küçük, eşit ($<$, $>$, $=$).

$O = \{op_1, \dots, op_m\}$: Reel sayılar üzerindeki işlemler ($+$, $-$, $\times$, $/$).

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.8

Yazılım Tasarımı Kalitesi

Sezgisel (gerçek dünya) bağıntı sistemi (Empirical Relation System) Örnekleri

Örnek 1: Varlıklar: ipler, İlgilenilen nitelik: uzunluk

ip1 "daha uzun" ip2 "aynı uzunlukta" "uç uca eklenir"

Örnek 2: Varlıklar: yazılım sınıfları, İlgilenilen nitelik: karmaşıklık

class ClassA{
public:
ClassA(double);
void FirstMethod(int);
float SecondMethod();
private:
double firstAttribute;
int otherAttribute;
};

"daha karmaşık"
"eşit veya daha karmaşık"

Bu bağıntıları görmek (sezmek) ilk örnekteki gibi kolay değil.

class ClassB{
public:
ClassB();
int FirstMethod(int);
double SecondMethod();
private:
int otherAttribute;
};

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.9

Yazılım Tasarımı Kalitesi

Sezgisel bağıntı sistemi (Empirical Relation System) Örnekleri (devamı)

Sezgisel bağıntı sistemi belli bir niteliğe göre oluşturulur.

Örneğin iplerin uzunlukları yerine kalınlıkları ya da dayanaklıklarına göre de ayrı bağıntı sistemleri bulunabilir.

Örnek 3: Varlıklar: ipler, İlgilenilen nitelik: kalınlık

ip1 "daha ince" ip2 "aynı kalınlıkta" "birbirine sarılır"

Örnek 4: Varlıklar: yazılım sınıfları, İlgilenilen nitelik: anlaşılabilirlik

class ClassA{
};

"daha kolay anlaşılır"
"eşit veya daha kolay anlaşılır"

class ClassB{
};

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.10

Yazılım Tasarımı Kalitesi

Temsil koşulu (representation condition)

$E=(C, R^E, O^E)$ bir gerçek dünya bağıntı sistemidir (*Empirical Relation System*).

$F=(N, R^F, O^F)$ bir sayısal (formel) bağıntı sistemidir (*Numerical (formal) Relation System*).

Gerçek dünyadaki C varlık kümesindeki nesnelerin belli bir niteliğinin ölçülebilmesi için E'den F'ye aşağıdaki özelliklere sahip bir ölçme, yani karşı düşürme (*mapping*) işlemi M tanımlanmalıdır.

M: $(C, R^E, O^E) \rightarrow (N, R^F, O^F)$ (Sezgisel Dünya) $\rightarrow$ (Sayısal Dünya)

Bu ölçme işlemine (M) **temsil (representation)** denir.

Temsil M, C kümesindeki niteliklere N'den bir sayı düşürürken aynı zamanda, R^E kümesindeki bağıntılarla R^F kümesindeki bağıntıları ve O^E kümesindeki işlemlerle O^F kümesindeki işlemleri birbirlerine karşı düşürür.

Temsil koşulu (representation condition):

- $\forall x, y \in C$ ve " $<$ " $\in R^E$, $\exists$ " $<$ " $\in R^F$: $x < y \Leftrightarrow M(x) < M(y)$
- $\forall x, y \in C$ ve " $\oplus$ " $\in O^E$, $\exists$ " $+$ " $\in O^F$: $M(x \oplus y) = M(x) + M(y)$

Örnek: Gerçek dünyada ip1 ip2'den "daha kısaysa", ölçme sonucu ip1'e atanan sayı ip2'ye atanan sayıdan "daha küçüktür". Bu ters yönde de doğrudur.

İki ip birbirine eklenirse ölçme değerleri toplanır.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.11

Yazılım Tasarımı Kalitesi

Örnek: Temsil koşulu (representation condition):

İşlemleri (*operation*) dikkate almadan sadece varlıklar ve bağıntılar ile de bağıntı sistemleri oluşturulabilir. $E=(C, R^E)$, $F=(N, R^F)$

Nitelik: Uzunluk
Varlıklar: ip

ip1 ip2 ip3

R^E :
"daha kısa"
"daha uzun"
"eşit"

Temsil (Representation)

R^F :
" $<$ "
" $>$ "
" $=$ "

$E=(C, R^E)$ M $F=(N, R^F)$

Gerçek dünya (sezgisel) bağıntı sistemi (Empirical Relation System)

Sayısal (formel) bağıntı sistemi (Numerical (formal) Relation Sys.)

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.12

Yazılım Tasarımı Kalitesi	
Tekli (unary) ilişkiler Tek bir varlığın niteliği üzerinde tanımlanan tekli (birli) ilişkiler de vardır. Örneğin, "A kişinin boyu çok uzundur". Matematiksel dünyada bu ilişkiler eşik değerleri kullanılarak temsil edilir. Örneğin, $M(A) > 1.90m$. Bunun (M) geçerli bir temsil olması için boyu 1.90m'den uzun olanların o toplumda "çok uzun" olarak kabul edilmesi gerekir. Birli ilişkiler yazılım dünyasında belli kararları verebilmek için gereklidirler. Örneğin, "A sınıfı fazla büyük, bölünmesi gerekir", "A sınıfı fazla karmaşık, yeniden tasarlanması gerekir", "A sınıfının anlaşılabilirliği kabul edilebilir düzeydedir, düzenleme yapmaya gerek yoktur". Buradaki temel zorluk temsil koşulunu sağlayan eşik değerlerini belirleyebilmektir. Bu konu "Bölüm 6: Tasarım Kusurlarının Belirlenmesi" bölümünde ele alınmıştır.	
akademi.itu.edu.tr/buzluca www.buzluca.info	2012 - 2022 Feza BUZLUCA 3.13

Yazılım Tasarımı Kalitesi	Lisans: https://creativecommons.org/licenses/by-nc-nd/4.0/deed.tr
Ölçmenin Temel Adımları - Gerçek dünya varlığının ölçmek istenen niteliği tanımlanır. Niteliklerin geçerli bir modeli (formül, denklem de olabilir) oluşturulur. Örneğin bir yazılım sınıfın büyüklüğünün tanımı veya bir yazılım modülünün karmaşıklığının tanımı. - Ölçülecek nitelik için gerçek dünyada geçerli olan sezgisel ilişkiler tanımlanır. Ölçülmek istenen nitelik ile ilgili olarak gerçek dünyada geçerli olan (bilinen, uzlaşılmış olan) ilişkiler nelerdir? Örneğin: eşit, aynı değil, daha büyük, çok karmaşık. - Sezgisel ilişkilere karşı düşen formel (sayısal) ilişkiler belirlenir. Örneğin $a = b$, $a \neq b$, $a > b$, $a > 75$ - Gerçek dünya varlıklarının niteliklerine sayılar karşı düşürülür. $M : E \rightarrow F$ (Sezgisel Dünya) $\rightarrow$ (Sayısal Dünya) dönüşümü tamamlanmış olur. - Temsil koşulunun sağlanıp sağlanmadığı kontrol edilir. Sezgisel ve sayısal ilişkiler iki yönlü olarak birbirini gerektiriyor mu? Aynı E sezgisel sistemini temsil eden birden fazla F sayısal sistemi oluşturmak mümkündür (Bkz. Kabul edilebilir dönüşüm).	
akademi.itu.edu.tr/buzluca www.buzluca.info	2012 - 2022 Feza BUZLUCA 3.14

Yazılım Tasarımı Kalitesi	
Ölçmede Niteliklerin Modeli: Ölçülmek istenen nitelik doğru bir biçimde tanımlanmalı ve modellenmeli. Örnekler: - Bir kişinin boyu ölçülmeden önce "kişinin boyu" niteliği modellenmeli (tanımlanmalı). Ölçme sırasında ayakkabı giyilebilir mi? Saçlar boy uzunluğunun içinde midir? - Bir kişinin zeka düzeyi belirlenmek isteniyor. Zekanın tanımı nedir? Tek bir nitelik midir, alt niteliklerden mi oluşur? IQ testi sonucu bize doğru bilgi verir mi (temsil özelliği var mı)? - Bir yazılım modülünün (alt program, metot vb.) karmaşıklığı ölçülecek. Karmaşıklık nedir? Örnek: Karmaşıklık (sadece) McCabe sayısı ile tanımlamak doğru mudur? McCabe sayısı bir program modülünde kaç tane bağımsız yol olduğunu verir. McCabe sayısı programdaki karar noktalarına bağlıdır. Karmaşıklığın artmasıyla programın anlaşılabilirliğinin azalması ve bakımının zorlaşması beklenir. Ancak bazı programlarda karar noktaları arttıkça programın daha kolay anlaşılabilirliği ve bakımının daha kolay yapıldığı görülmüştür (temsil özelliği?).	
akademi.itu.edu.tr/buzluca www.buzluca.info	2012 - 2022 Feza BUZLUCA 3.15

Yazılım Tasarımı Kalitesi	
Ölçek (Scale) $E=(C,R^E,O^E)$ bir gerçek dünya bağıntı sistemi (Empirical Relation System), $F=(N,R^F,O^F)$ bir sayısal bağıntı sistemi (Numerical (formal) Relation System) ve $M: E \rightarrow F$, şeklinde bir ölçü (temsil) olmak üzere (E, F, M) üçlüsüne ölçek (scale) denir. Örnekler: Kişilerin boyunun santimetre ile ölçülmesi bir ölçek, inç olarak ölçülmesi başka bir ölçektir. Bir yazılımın boyutunun satır sayısı ile ölçülmesi, sınıf sayısı ile ölçülmesi. Gerçek dünyadaki hangi nitelikler ölçülebilir? E sezgisel sistemiyle tarif edilen bir niteliğin anlamlı olarak ölçülebilmesi için (E, F, M) şeklinde ölçek oluşturabilecek bir ölçüm (measure) M bulunabilmesi gerekir. Diğer bir deyişle E'den F'ye temsil koşulunu sağlayan bir karşı düşürme (mapping) M bulunabilirse E'de tarif edilen varlıkların niteliği ölçülebilir. Hatırlatma: Temsil M, gerçek dünyada sezgisel (doğal) olarak belirlenen ilişkilerin sayısal sistemde de korunmasını sağlar. Ölçme değerleri arasında bulunan ilişkiler gerçek dünyada da geçerlidir.	
akademi.itu.edu.tr/buzluca www.buzluca.info	2012 - 2022 Feza BUZLUCA 3.16

Yazılım Tasarımı Kalitesi	
Kabul Edilebilir Dönüşüm (Admissible transformation) E sezgisel sisteminde yer alan bir nitelik birden fazla ölçekle ölçülebilir. Örneğin: kişilerin boyunun uzunluğu santimetre ölçeği ile sayısal sisteme dönüştürülebileceği (ölçülebileceği) gibi inç ölçeği ile de dönüştürülebilir. (E, F, M) bir ölçekse ve (E, F', M') aynı E sezgisel sistemine ilişkin başka bir ölçekse M' ölçümü (temsili) kabul edilebilir dönüşümdür (admissible transformation). Örneğin: inç ölçeğindeki sonuçlar santimetre ölçeğine dönüştürülebilir. $T(x) = 2.54 \cdot x$ M' ölçümü, M ölçümünün E sisteminde sağladığı tüm ilişkileri sağlamalıdır. Örnek: Gerçek dünyada ip1, ip2'den daha uzunsa tüm kabul edilebilir dönüşümlerde ip1'e atanan sayı, ip2'ye atanan sayıdan daha büyüktür. Sezgisel sistemde: ip1 "daha uzun" ip2 $\Rightarrow M(ip1) > M(ip2)$ (santimetre ölçeği) $\Rightarrow M'(ip1) > M'(ip2)$ (inç ölçeği) Bu örnekte iki ölçek arasında $M = cM'$ şeklinde bir bağıntı (dönüşüm) vardır. Ölçekler arasında tiplerine bağlı olarak farklı bağıntılar (dönüşüm yöntemi) olur.	
akademi.itu.edu.tr/buzluca www.buzluca.info	2012 - 2022 Feza BUZLUCA 3.17

Yazılım Tasarımı Kalitesi	
Ölçek Tipleri (Scale Types) Ele alınan E sezgisel sistemindeki - ilişkilerin çeşitliliğine (sayısına) ve - bu ilişkilerin ne kadarının ölçme işleminde sağlandığına (temsil edildiğine) bağlı olarak, bu sezgisel sistemin ölçülmesinde kullanılan ölçeklerin ve kabul edilebilir dönüşümlerin nitelikleri değişir. Temsil edilen bağıntıların ayrıntılarına bağlı olarak ölçekler 5 farklı tipe ayrılır: 1. İsimlendirme (Nominal) Ölçeği: - Sadece etiketleme (isimlendirme) (labeling) yapılır. - Varlıkların farklı niteliklerine farklı sayılar (veya simgeler) karşı düşürülür. - Atanan sayılar arasında sıralama amaçlı karşılaştırma yapmak mümkün değildir. - Sezgisel sistemdeki tek bağıntı "farklı olma" (veya "aynı olma") bağıntısı olabilir. - Atanan sayılar arasında aritmetik işlem yapmak anlamlı değildir. Örnekler: - Programlarda fonksiyonların döndürdükleri hata kodları: dosya açılmadı: 1, dosya okunamıyor: 2, disk dolu: 3 vs. - Kişilerin etiketlenmesi: kadın=1, erkek=0. Farklı niteliklere farklı değerler atayan tüm yöntemler kabul edilebilir dönüşümlerdir.	
akademi.itu.edu.tr/buzluca www.buzluca.info	2012 - 2022 Feza BUZLUCA 3.18

Yazılım Tasarımı Kalitesi

2. Sırasal (Ordinal) Ölçek:

- Varlıklara atanan sayılar belli bir niteliğe göre varlıklar arasında (sadece) sıralama yapılmasını sağlar.
- Geçerli bağıntılar: $A < B$, $A > B$, $A = B$, $A \neq B$.
- Geçersiz bağıntılar: "A, B'den 5 birim daha büyük" ya da "A, B'nin 2 katıdır" gibi önermeler yapılamaz.
- Atanan sayılar arasındaki farklar (aralıklar) anlamlı değildir.
- Atanan sayılar arasında aritmetik işlem yapmak anlamlı değildir.

Örnekler:

- Mohs mineral sertlik katsayısı; bir malzemenin değerini çizelbesi ilkesine dayanır. A:2, B:1 ise A, B'den daha serttir.
- Hava kalitesine (ya da hava kirliliğine) atanan sayılar. 1: Düşük risk, 2: Orta risk, 3: Yüksek risk, 4: Çok yüksek risk
- Hatlara atanan değerler: önemsiz (0), kritik(1), çok kritik(2).

Aynı sırayı koruyan temsiller kabul edilebilir dönüşümlerdir. Monotonik artan fonksiyonlar kabul edilebilir dönüşümlerdir.

$M(x) > M(y) \Rightarrow M'(x) > M'(y)$

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.19

Yazılım Tasarımı Kalitesi Lisans: <https://creativecommons.org/licenses/by-nc-nd/4.0/deed.tr>

3. Aralık (Interval) Ölçeği:

- Varlıkların niteliklerine atanan sayılar, varlıklar arasında hem sıralama yapılmasını sağlarlar hem de sayılar arasındaki farklarda (aralıklar) anlamlıdır. Örneğin sıcaklığın $10^{\circ}C$ 'den $20^{\circ}C$ 'ye çıkmasıyla $30^{\circ}C$ 'den $40^{\circ}C$ 'ye çıkması aynı sıcaklık farkının oluştuğu anlamına gelir.
- Geçerli bağıntılar: $A - B = C - D$, $A - B > C - D$ gibi.
- Geçersiz bağıntı: "A, B'nin 2 katı" gibi önermeler yapılamaz.
- Çünkü bu tür sistemlerde mutlak sıfır noktası yoktur. Yani sıfır değeri hiçlik (yokluk) belirtmez.
- Sayılar arasındaki aralıklar (farklar) anlamlıdır ancak oranlar geçerli değildir.
- Atanan sayılar arasında toplama ve çıkarma işlemleri yapılabilir; ancak çarpma ve bölme yapılamaz (anlamlı değildir).

Örnek:

- Sıcaklığın Celsius, Fahrenheit ölçekleri ile ölçülmesi.

Kabul edilebilir dönüşümler: $M' = a \cdot M + b$; $a > 0$

Örnek: $^{\circ}F = 1.8 \cdot ^{\circ}C + 32$

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.20

Yazılım Tasarımı Kalitesi

4. Orantı (Ratio) Ölçeği:

- En üst düzeydeki ölçektir (en zengin ilişkiye sahip sistemi temsil eder).
- Ölçme sistemi mutlak sıfır değerine sahiptir. Bir niteliğe sıfır sayısının atanması yokluğu ifade eder.
- Aralık ölçeğine ek olarak atanan sayılar arasındaki oranlar da anlamlıdır. Örneğin "ip1'in uzunluğu ip2'nin uzunluğunun 2 katıdır" önermesi anlamlıdır.
- Atanan sayılara tüm aritmetik işlemler uygulanabilir.

Örnekler:

- Varlıkların uzunluk
- Nesnelerin ağırlıkları

Kabul edilebilir dönüşümler:

$M' = a \cdot M$; $a > 0$

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.21

Yazılım Tasarımı Kalitesi

5. Mutlak (Absolute) Ölçek:

- Varlıkların nitelikleri belli bir şeyin sayısı (adeti) olarak ifade edilir. Örnek: Programdaki satır sayısı, testlerde rastlanan hata sayısı dersi alan öğrenci sayısı
- Ölçme işlemi, ilgili niteliği oluşturan unsurlar sayılarak yapılır.
- Söz konusu niteliği sadece tek bir temsil ile ölçmek mümkündür. $M' = M$
- Diğer tüm ölçeklerin özelliklerine sahiptir (sıra, aralık, orantı ilişkileri sağlanır).

Dikkat edilmesi gereken noktalar:

- Sayma yapılan her ölçme işlemi mutlak ölçekte olmaz.
- Örneğin; bir program modülünün boyutu (uzunluğu) ölçülme istendiğinde programdaki satırlar sayılabilir. Ancak bu ölçme işlemi mutlak ölçekte değil, orantı ölçeğinde olur. Çünkü nitelik bir şeyin sayısı olarak tanımlanmamıştır ve başka şekillerde de ölçülebilir. Örneğin; karakter sayısı veya bellekte kapladığı yer olarak ölçülebilir. Eğer ölçülme istenen nitelik "programdaki satır sayısı" olarak tanımlanırsa ölçme işlemi mutlak ölçekte olur.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.22

Yazılım Tasarımı Kalitesi

Ölçeğin tipini belirleyen nedir?

- Ölçeğin tipi, sezgisel sistemdeki varlıkların nitelikleri arasındaki bağıntıların ayrıntılarına bağlıdır. Diğer bir deyişle varlıkların nitelikleri arasında ne gibi ilişkiler algılanabildiğine (sezilebildiğine) bağlıdır.
- Bu ilişkilerin hangilerinin ölçme sonucu belirlenmek istendiğine bağlıdır.

Günümüzde yazılımların birçok niteliği hakkındaki algımız en fazla karşılaştırma düzeyindedir.

Örneğin: "A programı, B programından daha karmaşıktır."

Daha doğru ifade: "A programının karmaşıklığı, B programının karmaşıklığından daha büyüktür."

"A sınıfının uyumu B sınıfının uyumundan daha düşüktür" gibi.

Bu tür nitelikler **sırasal ölçekle** ölçülebilirler.

Eğer iki program parçasını (A, B) karşılaştırıp "A, B'den 5 birim daha karmaşık" kararı verilebilse karmaşıklık aralık (interval) ölçeği ile ölçülebilirdi.

Gerçek dünyada "A, B'den 2 kat daha karmaşık" kararı verilebilseydi karmaşıklık orantı (ratio) ölçeği ile ölçülebilirdi.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.23

Yazılım Tasarımı Kalitesi

Geçerli olan ölçek, ölçme sonuçlarının nasıl değerlendirileceğini etkiler:

Ölçeğin tipi sonuçlar üzerinde hangi yorumların yapılabileceğini ve hangi istatistiksel işlemlerin uygulanabileceğini belirler.

a. İsimlendirme (nominal) ölçek:

Sadece A ve B'nin incelenen niteliklerinin aynı veya farklı olduğu söylenebilir. $M(A)=1$, $M(B)=2$ ise "A, B'den daha küçüktür" önermesi geçerli olmaz. Sonuçlar üzerinde istatistiksel olarak sadece yoğunluk hesabı yapılabilir.

Örneğin;

Bu sistemde kaç tane X değerine sahip nitelik var?

Hataların kaç tanesi "derleme", kaç tanesi "çalışma zamanı" tipinde?

Belli bir sürede olan olayların ne kadarı Y niteliğine sahip?

b. Sırasal ölçek:

A'nın niteliğinin B'nin niteliğinden küçük/büyük olduğu söylenebilir ama ne kadar küçük/büyük olduğu söylenemez.

Sonuçlar üzerinde istatistiksel olarak medyan hesabı yapılabilir.

Bu değerler üzerinde ortalama alınması yanlıştır, çünkü değerler arasındaki aralıklar (farklar) anlamlı değildir.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.24

Yazılım Tasarımı Kalitesi

Geçerli olan ölçek, ölçme sonuçlarının nasıl değerlendirileceğini etkiler: (devamı)

c. Aralık ölçeği:
 Ölçüm sonuçları arasındaki farklar da anlamlıdır.
 Bir değerin diğerinden ne kadar büyük olduğu söylenebilir.
 Ancak orantısal yorum yapılamaz.
 Ölçme değerinin sıfır olması o niteliğin "yokluğu" anlamına gelmez.
 Örneğin, sıcaklık ölçümünün sonucu 0° C elde edilmesi sıcaklık olmadığı anlamına gelmez.
 Sonuçlar üzerinde (öncekilere ek olarak) istatistiksel olarak aritmetik ortalama, standart sapma hesapları yapılabilir.

d. Orantı ölçeği (ve mutlak ölçek):
 Sonuçların birbirlerine oranları da anlamlıdır.
 Bir değerin diğerinden kaç kat kadar büyük olduğu söylenebilir.
 Ölçme değerinin sıfır olması o niteliğin "yokluğu" anlamına gelir.
 Örneğin uzunluğun veya ağırlığın sıfır olması.
 Sonuçlar üzerinde (öncekilere ek olarak) gerektiğinde geometrik ortalama hesabı yapılabilir.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.25

Yazılım Tasarımı Kalitesi Lisans: <https://creativecommons.org/licenses/by-nc-nd/4.0/deed.tr>

Ölçme sonuçları üzerinde yapılan yorumların anlamlı (meaningful) olması:

- Sıcaklık 10° C'den 20° C'ye çıktığında "sıcaklık iki kat arttı" yorumu anlamlı mıdır?
- Kütle 10 kg'den 30 kg'ye çıktığında "kütle üç kat arttı" yorumu anlamlı mıdır?
- Birinci ipin uzunluğu 10 inch ikinci ipin uzunluğu 20 inch ise "birinci ipin uzunluğu ikinci ipin uzunluğunun iki katıdır" yorumu anlamlı mıdır?

Bir ölçme sonucu üzerinde yapılan yorumun anlamlı olması demek, bu yorumun **gerçek dünyada geçerli** (temsil prensibine uygun) olması demektir.
 Örneğin; sıcaklık 10° C'den 20° C'ye çıktığında "sıcaklık iki kat arttı" yorumu anlamlı ise bu sıcaklık değişiminin gerçek dünyadaki etkileri de bu orana uygun olmalı.

Ölçme sonuçları üzerinde yapılan yorumun anlamlı olması **o ölçmenin hangi ölçekte olduğuyla bağlıdır**.

Ölçme sonucunun **doğru** olması ile anlamlı olarak yorumlanması farklı konulardır.
 Bir yorumun anlamlı olup olmadığını sınamak için, bu yorumun doğru veya yanlışlığının ölçme yapılan sistemin tüm kabul edilebilir dönüşümlerinde aynı olup olmadığı kontrol edilebilir.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.26

Yazılım Tasarımı Kalitesi

Ölçme sonuçları üzerinde yapılan yorumların anlamlı olması (devamı)

Örnek 1:
 "Sıcaklık 2 katına çıktı" yorumu anlamlı mıdır?
 Celsius ve Fahrenheit birimlerinde ölçülen sıcaklık, aralık ölçeğinde olduğundan bu **yorum anlamlı değildir**.

Ölçeği bilmeden de bu yorumun anlamlı olup olmadığını aşağıdaki yöntemle sınayabiliriz.
 Yorumun anlamlı olabilmesi için bu önermenin sonucunun (doğru/yanlış) o anda kullanılan sıcaklık ölçeğinin tüm kabul edilebilir dönüşümlerde aynı olması gerekir.
 $10^{\circ}C = 50^{\circ}F$
 $20^{\circ}C = 68^{\circ}F$
 Celsius birimindeki değer iki katına çıkmasına karşın Fahrenheit birimindeki değer iki katına çıkmamıştır.
 Bu sinamaya göre "sıcaklık 2 katına çıktı" yorumu anlamlı değildir.
 Bu ölçmede aralık ölçeği geçerli olduğundan "Sıcaklık 10° C arttı" yorumu anlamlıdır.
 Celsius biriminde 10° C'lik fark Fahrenheit ölçeğinde her zaman 18° F'lik farka karşılık gelecektir.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.27

Yazılım Tasarımı Kalitesi

Ölçme sonuçları üzerinde yapılan yorumların anlamlı olması (devamı)

Örnek 2:
 "Birinci ipin boyu ikinci ipin boyunun 2 katıdır" yorumu anlamlı mıdır?
 Uzunluk ölçümünde ölçek tipi orantı (ratio) ölçeği olduğundan **yorum anlamlıdır**.

Sinama:
 Birinci ip: 10 inch = 25.4 cm
 İkinci ip: 20 inch = 50.8 cm
 Her iki birimde de önerme doğru olduğundan bu sinamaya göre de yorum anlamlıdır.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.28

Yazılım Tasarımı Kalitesi

Ölçme sonuçları üzerinde yapılan yorumların anlamlı olması (devamı)

Örnek 3: İki yazılım sisteminin (X ve Y) modüllerinin (örneğin sınıf) anlaşılabilirlikleri (understandability) (daha doğrusu anlaşılma zorlukları) değerlendiriliyor ve her birine aşağıdaki beş dereceden biri atanıyor.
 Çok basit (trivial), Basit (simple), Orta (moderate), Karmaşık (Complex), Anlaşılmaz (Incomprehensible)

Bu dereceler arasındaki aralıklar belirli (ve eşit) değildir. Bu nedenle bu derecelendirmeye yapılan ölçme **sıralama ölçeğinde** olur.
 Buna göre yukarıdaki dereceleri temsil etmek üzere aşağıdaki sayılar kullanılabilir.

	Çok basit	Basit	Orta	Karmaşık	Anlaşılmaz
M :	1	2	3	4	5
M' :	1	2	3	4	10

(kabul edilebilir dönüşüm)

X sistemindeki 5 adet modüle (x_i), değerlendiriciler aşağıdaki dereceleri atamıştır:
 x_1 : Çok basit, x_2 : Basit, x_3 : Basit, x_4 : Orta, x_5 : Anlaşılmaz

Y sistemindeki 7 adet modüle (y_i) ise aşağıdaki dereceler atanmıştır:
 y_1 : Basit, y_2 : Orta, y_3 : Orta, y_4 : Orta, y_5 : Karmaşık, y_6 : Karmaşık, y_7 : Karmaşık

Bu durumda iki sistemin anlaşılabilirliği nasıl karşılaştırılır? Ortalama almak uygun mudur?

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.29

Yazılım Tasarımı Kalitesi

Örnek 3: (Devamı)
 İki yazılım sisteminin (X ve Y) anlaşılabilirlikleri anlamlı olarak nasıl karşılaştırılabilir?

Ortalama üzerinden karşılaştırma:
 Eğer karşılaştırma için ortalamadan yararlanmak düşünülürse ve önceki yansıda verilen M ölçmesi kullanılırsa X sisteminin anlaşılma zorluğunun ortalaması 2,6; Y sisteminin ortalaması ise 3,1 (daha yüksek) olarak hesaplanır.

$$E[M(Y)] > E[M(X)]$$

$E[M(X)]$: M ölçmesinin X sistemine uygulanmasıyla edilen sonuçların beklenen değeri (ortalaması).

Eğer M' ölçmesi kullanılırsa X sisteminin anlaşılma zorluğunun ortalaması 3,6; Y sisteminin ortalaması ise 3,1 (daha düşük) olarak hesaplanır.

$$E[M'(X)] > E[M'(Y)]$$

Sıralama ölçeğinde ortalama anlamlı olmadığından iki sistemi ortalama üzerinden karşılaştırmak mümkün değildir.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.30

Yazılım Tasarımı Kalitesi

Örnek 3: (Devamı)

İki yazılım sisteminin (X ve Y) anlaşılabilirlikleri anlamlı olarak nasıl karşılaştırılabilir?

Medyan (merkezi değer) üzerinden karşılaştırma:

Her iki ölçme için de (M ve M') X sisteminin anlaşılma zorluğunun medyanı 2; Y sisteminin medyanı ise 3 (daha yüksek) olarak hesaplanır.

$Med[M(Y)] > Med[M(X)]$

$Med[M(X)]$: M ölçmesinin X sistemine uygulanmasıyla edilen sonuçların orta değeri (medyanı).

Örnekte verilen sistemler için her zaman $Med[M(X)]$ = basit, $Med[M(Y)]$ = orta bulunur.

Bu nedenle tüm kabul edilebilir dönüşümlerde aynı sonuç elde edilir.

Örnek M'' ölçmesi:

M'': Çok basit: 0,5 Basit: 3,8 Orta: 72 Karmaşık: 104 Anlaşılmaz: 500

$Med[M''(Y)] = 72 > Med[M''(X)] = 3,8$

Sıralama ölçeğinde medyan anlamlı olduğundan iki sistemi medyan üzerinden karşılaştırmak mümkündür.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.31

Yazılım Tasarımı Kalitesi

Ölçme değerlerinin (ölçülerin) elde edilmesi:

1. Doğrudan ölçme (*Direct measurement*):

Bazı niteliklerin ölçümleri başka niteliklere gerek duyulmadan doğrudan ölçülebilir.

Yazılımın iç nitelikleri genellikle doğrudan ölçülebilir.

2. Dolaylı ölçme (*Indirect measurement*):

Bazı niteliklerin değerleri tek bir ölçme işlemi ile diğer niteliklerden bağımsız olarak atanamaz.

Bu tür niteliklerin değerleri başka niteliklerin ölçümlerinden elde edilen değerlerin belli bir modele göre işlenmesiyle elde edilir. Hesaplama da (*calculation*) denir.

Yazılım yüksek düzeyli kalite kriterleri genellikle dolaylı olarak ölçülebilir.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.32

Yazılım Tasarımı Kalitesi

Nesnel (Objective) ve Öznel (Subjective) Ölçme

Nesnel (objective) ölçmenin sonuçları ölçmede rol oynayan kişilerden bağımsızdır. Aynı varlığın, aynı niteliği farklı kişiler tarafından da ölçülse aynı sonuçlar elde edilir.

Sonuçlar tutarlı olduğundan (eğer mümkünse) nesnel ölçme tercih edilir.

Öznel (subjective) ölçmenin sonuçları ölçmede rol oynayan kişilerin öznel (kişisel) görüşlerinden etkilenir.

Aynı varlığın, aynı niteliği için farklı kişiler farklı değerlendirmede bulunabilirler. Sonuçlar kişilere göre farklılık gösterse de çoğunluğun görüşü dikkate alındığında ortaya belli bir eğilim çıkıyorsa öznel ölçme sonuçları da yararlı olurlar.

Yazılım konusunda, özellikle geliştirilen nesnel ölçme yöntemlerinin sağlamlasını yapmak (doğruluğunu göstermek) için öznel ölçme sonuçlarından yararlanılır.

Örneğin, bir kodun anlaşılabilirliğini ölçebilen bir yöntem geliştirdiğiniz düşünün. Bu yöntemle değerlendirdiğiniz yazılım modüllerini belli sayıda programcıya okutarak anlaşılabilirliklerini puanlamalarını istersiniz.

Eğer yönteminizin değerlendirmesiyle programcıların değerlendirmesi arasında bir tutarlılık varsa sizin yönteminizin doğru sonuçlar ürettiğini iddia edebilirsiniz.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.33

Yazılım Tasarımı Kalitesi Lisans: <https://creativecommons.org/licenses/by-nc-nd/4.0/deed.tr>

Nesnel (Objective) ve Öznel (Subjective) Ölçme (devamı)

Örnek: Öznel (subjective) ölçme

Bir yazılım projesinde test senaryoları hazırlanmadan önce ilgili istemlerin (*requirements*) anlaşılabilirliklerinin test takımı tarafından değerlendirilmesi isteniyor. Takımın her üyesi her istem için 1 - 5 arası puan veriyor.

1: "Bu istemi tamamen anlamadım. Bu istemin karşılanıp karşılanmadığını belirleyecek test senaryolarını yazabilirim."

5: "Bu istemi anlamadım. Bu isteme ilişkin test senaryosunu yazamam."

Aşağıdaki tabloda kaç kişinin hangi tip istemlere hangi puanı verdiği gösterilmiştir.

	1 (iyi)	2	3	4	5 (kötü)
İster Tipi					
Performans istemleri	12	7	2	1	0
Veri tabanı istemleri	16	12	2	0	0
Arayüz istemleri	3	4	6	7	1
Diğer istemler	14	10	1	0	0

Öznel ölçme sonuçları:

Öznel sonuçlar arasında farklılıklar olsa da bu tabloya baktığımızda arayüz istemlerinin çok açık olmadığı ve gözden geçirilmeleri gerektiği sonucuna varabiliriz.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.34

Yazılım Tasarımı Kalitesi

Çok Bileşenli Nitelikler

Bazı nitelikler birden fazla bileşenden (alt nitelikten) oluşurlar.

Örneğin ISO/IEC kalite modelleri.

Çok bileşenli niteliklerin ölçülmesi, sonuçların değerlendirilmesi ve gerektiğinde varlıkların karşılaştırılabilmesi için birden fazla bileşenin (alt niteliğin) birlikte değerlendirilebilmesi için amaca uygun yöntemlerin geliştirilmesi gerekir.

Örnek:

Farklı ulaşım araçlarının kalitesinin değerlendirilmesi ve bu araçların karşılaştırılması isteniyor.

Araçların kalitesi iki bileşenden oluşuyor: yolculuk süresi ve maliyet.

Ulaşım Aracı	Yolculuk Süresi (saat)	Maliyet (TL)
Otomobil	3	30
Tren	5	60
Uçak	3,5	85
Otobüs	7	90

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.35

Yazılım Tasarımı Kalitesi

Çok Bileşenli Nitelikler (devamı)

Örnek (devamı):

Ulaşım aracı A'nın kalitesinin B ulaşım aracının kalitesinden yüksek olması için A'nın hem yolculuk süresinin hem de maliyetinin B'den düşük olması istenebilir.

Kalite (A) > Kalite (B): Eğer yolculuk süresi (A) < yolculuk süresi (B) VE maliyet (A) < maliyet (B)

Bu değerlendirme modeli kullanıldığında araçlar arasında kısmi sıralama bağıntısı (*partial order relation*) oluşur.

Bu modele göre otomobilin kalitesi diğer tüm araçlardan daha yüksektir. Otobüsün kalitesi diğer tüm araçlardan daha düşüktür. Ancak tren ve uçak arasında karşılaştırma yapmak mümkün değildir.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.36

Yazılım Tasarımı Kalitesi

Çok Bileşenli Nitelikler (devamı)

Çoklu alt niteliklerin tek bir sayı ile ifade edilmesi:

Tüm varlıkların birbirleriyle karşılaştırılabilmesi için birden fazla alt nitelikten elde edilen verilerin tek bir sayısal değerle ifade edilmesi istenebilir.

Temsil özelliğini (gerçek dünyaya uyumu) sağlayarak bu tür karmaşık kalite niteliklerinin tek sayı ile ifade edilmesi her zaman mümkün olmayabilir.

Bununla beraber, pratik yararlar sağlayan (örneğin sorunlu yazılım birimlerini büyük oranda belirleyen) modeller kurulabilir.

Çok sayıda alt nitelikten oluşan bir kalite niteliğine değer atamak üzere model oluştururken aşağıdaki iki sorunun çözülmesi gerekir.

1. Alt niteliklere ilişkin değerlerin aralıkları farklıdır.

Örneğin, yolculuk süresi 3 - 7 saat arasında değişirken, maliyet 30 - 90 TL arasında değişmektedir.

Farklı aralıklarda değişen değerleri aynı denklemde kullanmak sorunlara neden olur.

Bu sorunu çözmek için, değerler uygun yöntemlerle aynı aralığa getirilir.

- Örneğin; min-max normalizasyonu. $y = (x - \min) / (\max - \min)$
- Örneğin; alt niteliklerden en küçüğünün değeri 1 kabul edilerek diğer alt niteliklerin değerleri orantısız olarak hesaplanabilir.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.37

Yazılım Tasarımı Kalitesi

Çoklu alt niteliklerin tek bir sayı ile ifade edilmesi (devamı):

1. Alt niteliklere ilişkin değerlerin aralıkları farklıdır (devamı).

Alt niteliklerin, en küçük değer 1 kabul edilerek normalize edilmesi:

Ulaşım Aracı	Yolculuk Süresi (normalize)	Maliyet (normalize)
Otomobil	1	1
Tren	1,67	2
Uçak	1,17	2,83
Otobüs	2,33	3

Bu işlemin yapılabilmesi için bu alt niteliklerin orantı ölçeğinde değer alması gerekir. Ölçüm hangi ölçekteyse, normalizasyon işlemi ona uygun bir kabul edilebilir dönüşüm şeklinde olmalı.

Normalizasyon sonrası elde edilen değerler, orijinal değerler ile aynı ilişkileri ölçeğe uygun şekilde sağlamalı.

Bölüm 5'te ele alınacak olan örnek modelde (QMOOD) benzer bir normalizasyon işlemi yapılmıştır.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.38

Yazılım Tasarımı Kalitesi

Çoklu alt niteliklerin tek bir sayı ile ifade edilmesi (devamı):

2. Alt niteliklerin ilgili kalite niteliği açısından öneminin (ağırlığının) belirlenmesi

Örneğin yolculuk süresi ve maliyet eşit önemdeyse bir A aracının kalitesi normalize değerler kullanılarak aşağıdaki gibi hesaplanabilir:

Kalite (A) = Normalize Yolculuk Süresi (A) + Normalize Maliyet (A)

Bu modelde elde edilen daha küçük değerler daha yüksek kaliteyi ifade eder.

Daha büyük değerlerin daha yüksek kaliteyi ifade etmesi için formül aşağıdaki gibi oluşturulabilir:

Kalite (A) = 100 / (Normalize Yolculuk Süresi (A) + Normalize Maliyet (A))

Bu modele göre (her iki formüle göre de) tren yolculuğunun kalitesi uçak yolculuğundan yüksektir.

Kalite hesabında kullanılan metrikler tablodaki en küçük değer '1' kabul edilerek normalize edildiğinden, elde edilen kalite sonuç değerleri (örneğin Uçak: 25), kendi başlarına bir anlam ifade etmezler.

Bu değerler, tabloda yer alan araçların birbirlerine göre bağlı olarak değerlendirilmelerini sağlar.

Ulaşım Aracı	Kalite
Otomobil	2
Tren	3,67
Uçak	4
Otobüs	5,33

Ulaşım Aracı	Kalite
Otomobil	50
Tren	27,25
Uçak	25
Otobüs	18,76

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.39

Yazılım Tasarımı Kalitesi

2. Alt niteliklerin öneminin (ağırlığının) belirlenmesi (devamı)

Örneğin yolculuk süresinin, maliyetten iki kat daha önemli (ağırlıklı) olduğu düşünülebilir.

Ağırlıkların atanmasında gerçek dünyaya uyum (temsili) büyük önem taşır.

Alt niteliklerin ağırlıklarının birbirlerine oranı gerçek dünyada geçerli olmalı.

Yeni oranlara göre A aracının kalitesi aşağıdaki gibi hesaplanır:

Kalite (A) = 2 x Normalize Yolculuk Süresi (A) + Normalize Maliyet (A)

veya

Kalite (A) = 100 / (2 x Normalize Yolculuk Süresi (A) + Normalize Maliyet (A))

Bu modele göre ise uçak yolculuğunun kalitesi tren yolculuğundan daha yüksektir.

Ulaşım Aracı	Kalite
Otomobil	33,3
Uçak	19,34
Tren	18,73
Otobüs	13,05

Ağırlıkların belirlenmesinde çeşitli yöntemler kullanılmaktadır:

Geçmişteki deneyimlerden yararlanılır, var olan projeler üzerinde deneme yanılma yöntemleri uygulanır, makine öğrenmesi yöntemleri kullanılır.

İlerleyen bölümlerde bu yöntemler incelenecektir.

akademi.itu.edu.tr/buzluca
www.buzluca.info

2012 - 2022 Feza BUZLUCA 3.40