



Université Claude Bernard



Lyon 1

# Rapport

## TP Raisonnement à partir de cas

*Réalisé par :*

*CHBAB Khalid*

*ASRI Mohamed Amin*

## Contexte

Ce TP consiste à réaliser un algorithme qui prédit le prix d'un **appartement lyonnais** en se basant sur des problèmes antérieurs déjà résolus. Celui-ci est inspiré du raisonnement à partir de cas qui est une approche fondée sur l'expérience pour résoudre de nouveaux problèmes en adaptant des solutions précédemment réussies à des problèmes similaires. En abordant la mémoire, l'apprentissage, la planification et la résolution de problèmes, le raisonnement à partir de cas fournit une base pour une nouvelle technologie de systèmes informatiques intelligents capables de résoudre des problèmes et de s'adapter à de nouvelles situations. Dans notre cas, la réutilisation "intelligente" des connaissances issues de problèmes ou de cas déjà résolus repose sur le principe que plus deux problèmes sont similaires, plus leurs solutions seront similaires.

## 1. Descripteurs

La première étape dans notre raisonnement est le choix des descripteurs pertinents qui pourront décrire de façon très loyale notre espace de base de cas. La description d'un problème peut être sous forme de nombre (Superficie, Nombre de chambre, ...) ou bien une catégorie.

Pour notre problème, qui est la prédiction du prix de vente d'un appartement lyonnais, on s'est basé sur des descripteurs de pertinences différentes :

<i><b>Attribut</b></i>	<i><b>Type</b></i>	<i><b>Simple/Calculé</b></i>
Arrondissement	Symbol [String]	Simple
Superficie	Float	Simple
Nombre de chambres	Integer	Simple
Nombre de pièces	Integer	Simple
Etage	Integer	Simple
Standing	Integer	Simple
Prix	Float	Simple

Figure 1 : Descripteurs de cas

## 2. Base de cas

Après le choix de nos descripteurs, on a alimenté notre base de cas avec des données réels importé de différents sites de vente d'appartements type (seloger.com, logic-immo.com ...).

Dans un but pédagogique, nous considérons un cas comme une liste de descripteurs (ces descripteurs pouvant être des structures).

On a jugé qu'une base de 20 cas est suffisante pour notre situation. On les a classés de la manière suivante :

ID	Arrondissement	Superficie	Nombre de chambre	Nombre de pièces	Etage	Standing	Prix
1	69005	108,52	2	3	5	3	430 000
2	69004	61	3	4	4	3	390 000
3	69004	62,38	1	3	1	2	375 000
4	69006	59,4	1	2	2	2	348 000
5	69003	40,97	1	2	4	1	280 000
6	69008	70	2	3	4	1	264 000
7	69004	118,5	3	4	3	3	550 000
8	69002	78,41	3	4	2	3	420 000
9	69003	60,53	1	2	1	2	299 000
10	69100	83	2	3	4	3	385 000
11	69004	86,57	1	3	4	2	375 000
12	69003	85,5	3	4	1	3	482 500
13	69008	87,6	3	4	4	3	442 000
14	69008	93	4	5	5	3	547 000
15	69003	60,53	1	2	6	2	299 000
16	69007	69	2	3	3	3	400 000
17	69008	21	0	1	5	1	90 000
18	69004	145,66	4	5	2	3	670 000
19	69007	20,06	1	1	0	2	159 000
20	69003	25	0	1	1	1	115 000

Figure 2 : Base de cas

## 4. Remémoration

Il existe de nombreuses mesures de similarité prenant en compte les spécificités des descripteurs. Intuitivement, nous comprenons qu'il faut donner un poids élevé aux descripteurs de problèmes qui présentent des caractéristiques particulières et aussi aux descripteurs de problèmes qui présentent une forte dépendance avec les descripteurs de solutions et pour lesquels il n'existe pas d'opérateurs d'adaptation simples. Inversement, nous pouvons donner un poids faible aux descripteurs de problèmes présentant une faible dépendance.

Attributs	Types Attribut	Poid (coefficient)
Arrondissement	Symbol (String)	14
Superficie	Float	7
Nombre de chambres	Integer	6
Nombre de pièces	Integer	6
Etage	Integer	3
Standing	Integer	15
Prix	Integer	?

Figure 3 : Poids (Weight) de chaque descripteur

On remarque ici qu'on a donné des poids importants à deux descripteurs qui sont :

- Arrondissement
- Standing

On estime que ces deux descripteurs sont très pertinents dans la définition du prix de vente de l'appartement par rapport aux autres.

## 5. Adaptation

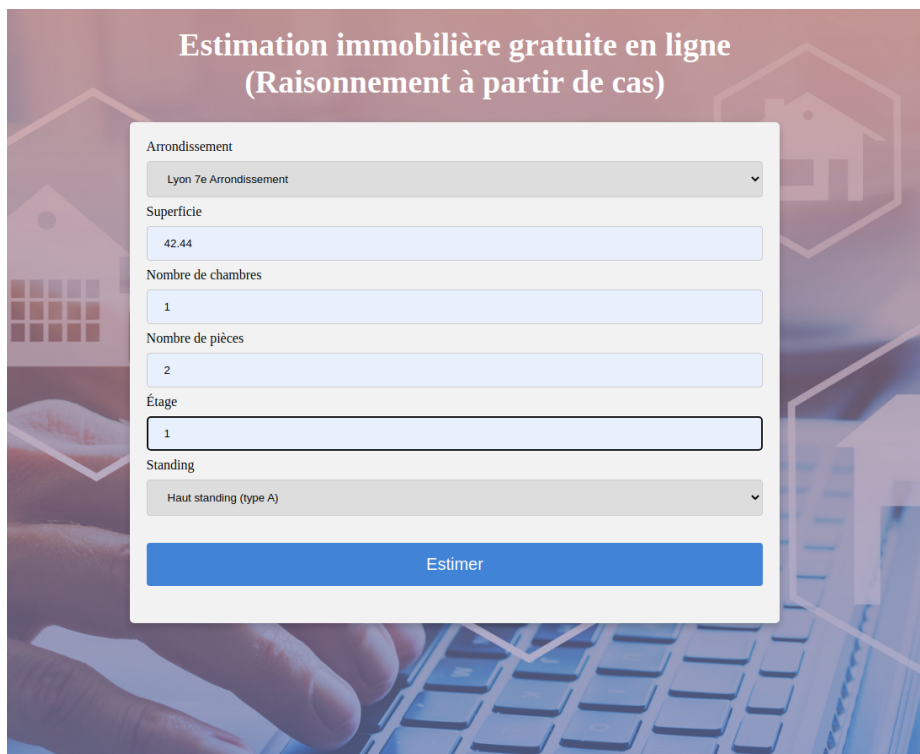
L'adaptation du cas source pour le problème cible est réalisée par la fonction `adapt(cas_cible,resultat)`. Le prix estimé est calculé en ajoutant une valeur  $\Delta_{price}$  au prix du cas source :

$$P_{estimate} = P_{source} + \Delta_{price}$$
$$\Delta_{price} = \sum_{n=d}^{descripteurs} (cas_{cible} - cas_{source}) * poids_d$$

À noter que pour des descripteurs contenant des valeurs de type *string*, le poids associé est une liste ordonnée.

## 6. Résultat

On a développé pour ce projet un site web où les utilisateurs peuvent renseigner des nouveaux cas cibles et le système de raisonnement à partir de cas recherche un cas source similaire et l'adapte pour afficher et donner une estimation de prix.



Estimation immobilière gratuite en ligne  
(Raisonnement à partir de cas)

Arrondissement  
Lyon 7e Arrondissement

Superficie  
42.44

Nombre de chambres  
1

Nombre de pièces  
2

Étage  
1

Standing  
Haut standing (type A)

Estimer

La page principale de l'application où l'utilisateur peut renseigner les descripteurs d'appartement

[« Retour](#)

	CAS SOURCE	CAS CIBLÉ
Arrondissement	Lyon 7e Arrondissement	Lyon 7e Arrondissement
Superficie (m <sup>2</sup> )	20.06 m <sup>2</sup>	42.44 m <sup>2</sup>
Nombre de chambres	1	1
Nombre de pièces	1	2
Étage	0	1
Standing	Moyen standing	Moyen standing
Prix (€)	159000.0 €	226140.0 €

La page d'affichage consiste d'un tableau de comparaison entre cas source similaire et cas de source cible et le prix estimé.

**Estimation immobilière gratuite en ligne**  
(Raisonnement à partir de cas)

Arrondissement  
Lyon 2e Arrondissement

Superficie  
62.44

Nombre de chambres  
2

Nombre de pièces  
2

Étage  
4

Standing  
Moyen standing (type B)

**Estimer**

Exemple d'un autre cas

« Retour

	CAS SOURCE	CAS CIBLÉ
Arrondissement	Lyon 3e Arrondissement	Lyon 2e Arrondissement
Superficie (m <sup>2</sup> )	60.53 m <sup>2</sup>	62.44 m <sup>2</sup>
Nombre de chambres	1	2
Nombre de pièces	2	2
Étage	6	4
Standing	Moyen standing	Moyen standing
Prix (€)	299000.0 €	274730.0 €

pour lancer le code il suffit d'installer les modules dans requirement.txt par

***pip install -r requirements.txt***

suivé par :

***flask run***

et après vous basculer vers localhost:5000/

la logique de raisonnement à partir de cas est dans le fichier main.py