

間取り図グラフの 利活用に関する研究

尾崎研究室

小黑 淳斗

上堀 健太

嶋野 友樹

アジェンダ

導入

- 間取り図とは
- 間取り図グラフとは

展開

- 間取り図グラフの利活用にむけて
 - ✓ 間取り図画像のグラフ化による検索支援に関する研究
 - ✓ 間取り図グラフ作成における誤認識・不認識の分析
 - ✓ グラフマイニングを用いた間取り図情報に基づく賃料分析

間取り図とは

間取り図とは

- 部屋の配置
- 洗面台やトイレ等を表す種々の記号
- LDK, WICなどの略語

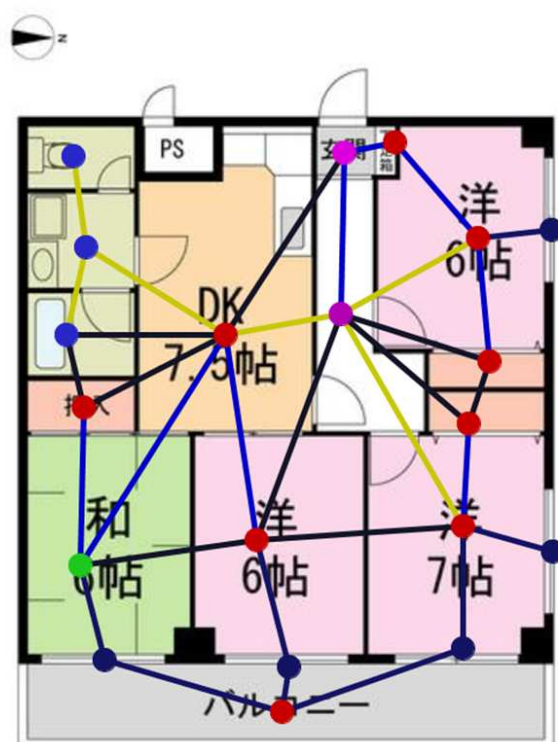
不均一で
様々な情報が
含まれている

- 洋室は3つ, 和室は1つ
- 玄関と下駄箱, 廊下, DKは隣接
など..



間取り図グラフとは

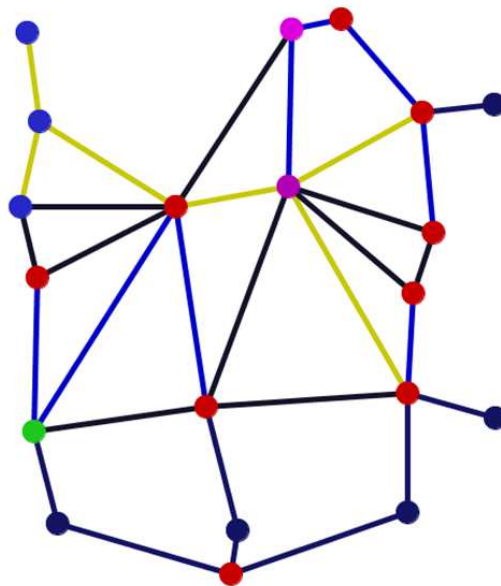
部屋の配置や繋がりに着目し、
間取り図をラベル付きグラフへ変換



部屋を頂点
繋がりを辺

間取り図グラフとは 利点

- 計算機での扱いが容易になる
- 構造的な側面からの特徴抽出や検索などへ展開できる



グラフとして
扱うことができる

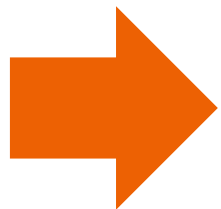
間取り図グラフとは 関連研究

- ・ 計画学におけるグラフ理論適用に関する研究：その2応用編
山中 知彦, 原 広司, 藤井 明, 渡辺 健一
日本建築学会論文報告集, Vol.342, pp.62-72 (1984)

間取り図を
無向グラフ
として表現

- ・ グラフマイニングを用いた室配置を考慮した賃料分析：京都市郊外の3LDKを中心とした賃貸マンションを対象として
瀧澤 重志, 吉田 一馬, 加藤 直樹
日本建築学会環境系論文集, Vol.73(623), pp.139-146 (2008)


間取り図を
隣接グラフ
として表現



間取り図をグラフ化することで
部屋の配置や繋がりに着目し, 構造的な側面から分析を行う


間取り図グラフとは ラベル種について

Node	Label
居室	洋室, 和室, リビング(L), ダイニング(D), LD, LK, DK, LDK
水回り	キッチン, トイレ, 風呂, 洗面室
収納	収納, 押入れ, ウォークインクローゼット(WIC)
その他	玄関, 廊下, ベランダ, 窓



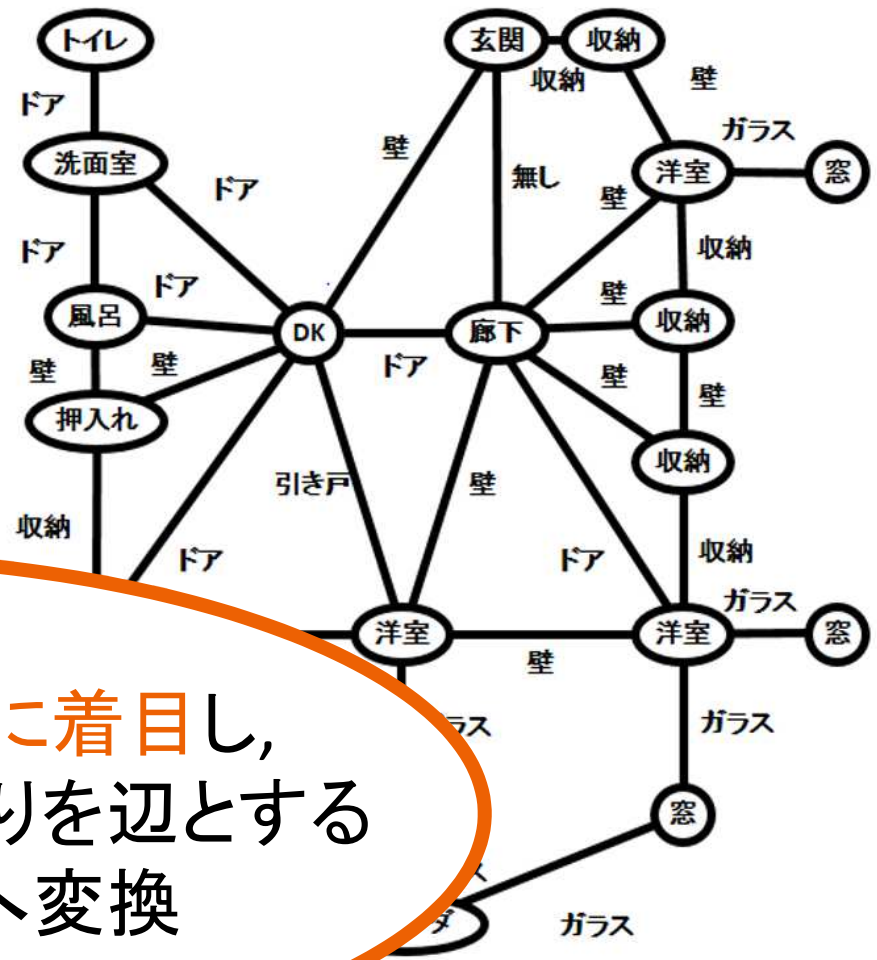
19種

Edge	Label
窓とつなぐエッジ	ガラス
収納とつなぐエッジ	収納
その他	ドア, 引き戸, 無し, 壁



6種

間取り図グラフへとは 具体例



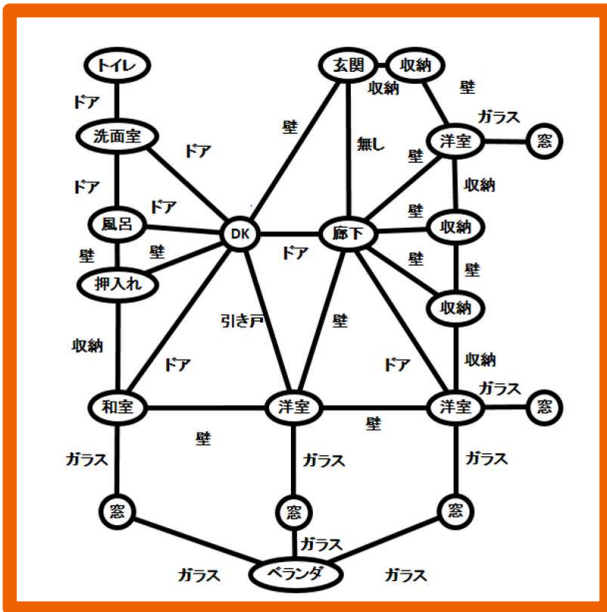
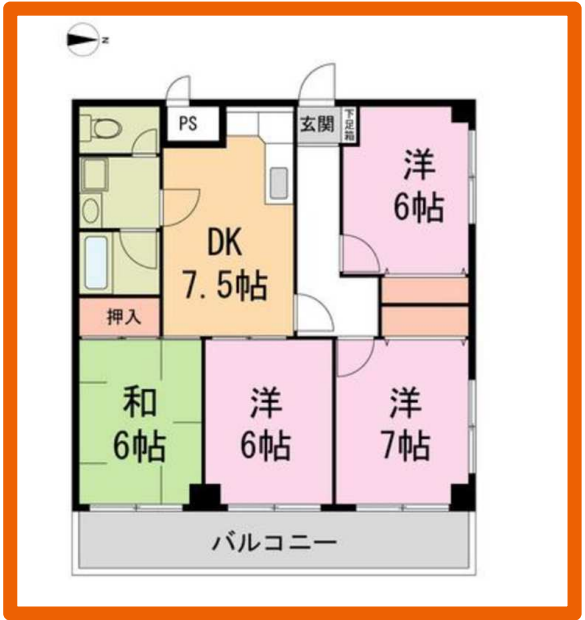
部屋の配置や繋がりに着目し、
部屋を頂点, そのつながりを辺とする
ラベル付きグラフへ変換

間取り図グラフの利活用にむけて

間取り図グラフの利活用に向けて

間取り図のグラフ化

検索の有用性



賃料分析



エラー傾向の分析



間取り図グラフ作成における 誤認識・不認識の分析

研究背景 目的

- 部屋探しにおける効率化の推進
- クラウドソーシングにおいて高品質なデータの獲得の可能性の向上

➡ 間取り図を見る際に起こしやすい間違いの特定や
その傾向を明らかにする



被験者が起こすエラーの分類 誤操作

- 誤操作

意図せぬノードやエッジを生成してしまった場合



うっかりノードを作成



※複数名の合議により, あらかじめ正解データを準備

被験者が起こすエラーの分類 誤認識

- 誤認識

(最終的に) 正解ラベルとは異なるラベルを付与した場合



ドア ×
引き戸 ○



※複数名の合議により, あらかじめ正解データを準備

被験者が起こすエラーの分類 不認識

- 不認識

本来あるべきノードやエッジが生成されなかった場合



窓の存在に
気づかなかった



※複数名の合議により, あらかじめ正解データを準備

データセットと正解データ

- 東京23区における部屋数が3部屋のマンション400件
- 被験者数10名, それぞれ40件を変換
- 正解データ

	数	標準偏差
Node	20.7	4.11
Edge	34.1	7.80

ノードでは窓
エッジでは壁
が最も多い

NodeLabel	平均数
窓	4.96
収納	4.16
洋室	2.19

EdgeLabel	平均数
壁	12.51
ガラス	7.38

提供：株式会社ネクスト, 国立情報学研究所

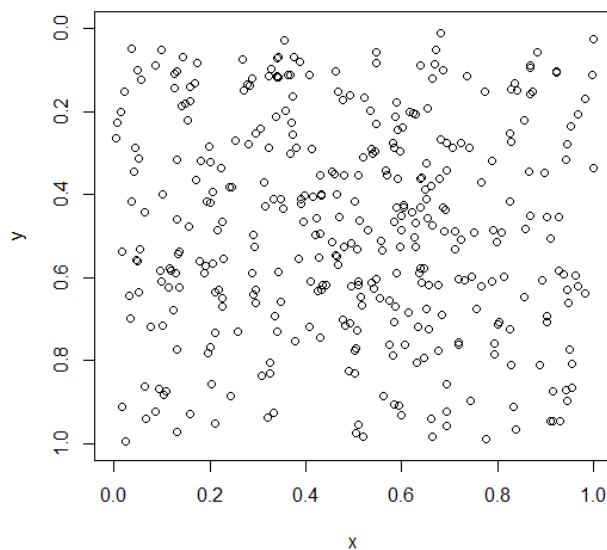
「HOME'Sデータセット」

住宅情報サイトHOME'Sに掲載された間取り図の画像データと物件の詳細情報

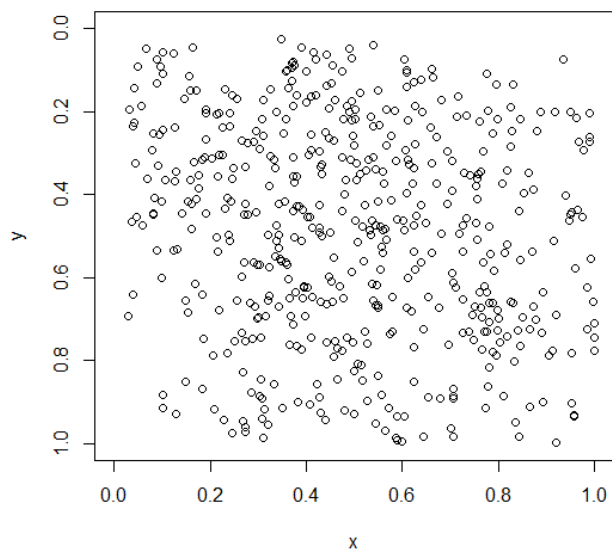
全体のエラー傾向の分析

- 間取り図あたりのグラフ変換時間 $t = 264.6$ 秒
- 相対座標：画像サイズに対するエラーの相対箇所

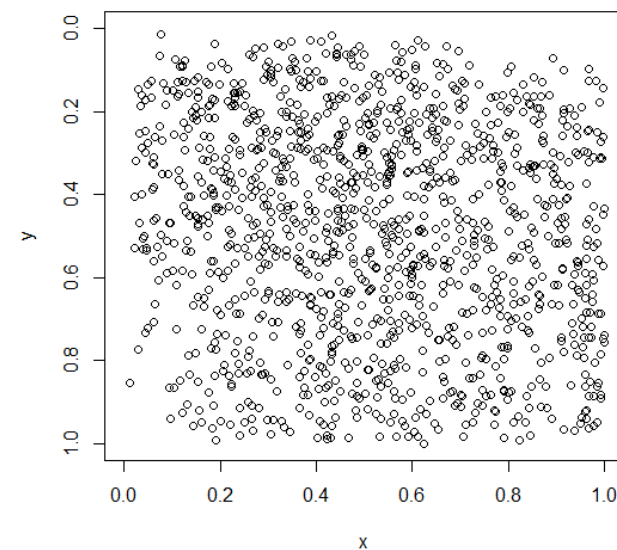
誤操作



誤認識



不認識



全体のエラー傾向の分析1

- 具体的な傾向を分析

相関ルール分析における確信度(条件付確率)の考え方を利用

A ⇒ **B** : ラベル**A**をラベル**B**に誤認識(不認識の場合, **B**=NULL)

確信度 $P(\mathbf{B} | \mathbf{A})$: ラベル**A**がある条件で**A**⇒**B**が起こる確率

ルールの確からしさの尺度を軸に抽出

全体のエラー傾向の分析結果1

correct label ⇒ wrong label	確信度
ダイニング⇒DK	0.2500
リビング⇒LDK	0.2500
LD⇒LDK	0.1675
引き戸⇒ドア	0.0685
LDK⇒LD	0.0682
ダイニング⇒LDK	0.0417
収納⇒WIC	0.0337
無し⇒壁	0.0317
LD⇒DK	0.0262
WIC⇒収納	0.0238
WIC⇒トイレ	0.0238
LDK⇒リビング	0.0227
ドア⇒引き戸	0.0202

correct label ⇒ wrong label	確信度
壁⇒NULL	0.2082
無し⇒NULL	0.1486
キッチン⇒NULL	0.1211
窓⇒NULL	0.0943
ガラス⇒NULL	0.0935
収納(Node)⇒NULL	0.0919
収納(Edge)⇒NULL	0.0893

全体のエラー傾向の分析2

- エラーと周辺状況との関連性を分析

相関ルール分析における確信度と支持度の考え方を利用

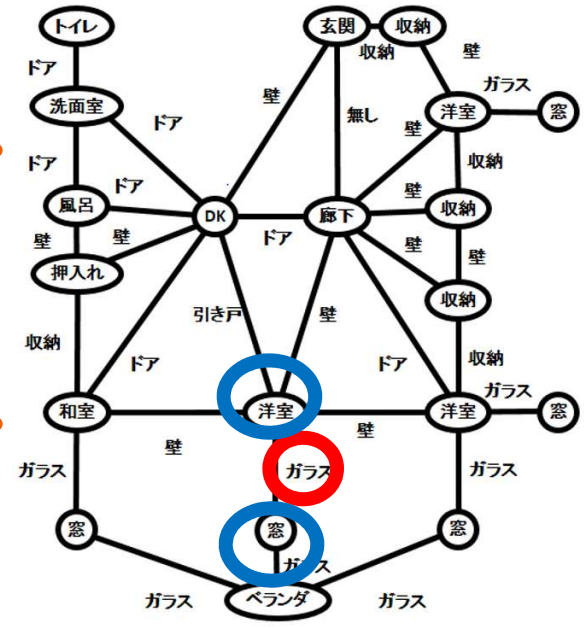
正解ラベルAと、その周辺状況S

確信度

支持度

玄関の周辺状況
 $S = \{ DK, 廊下, 収納 \}$

ガラスの周辺状況
 $S = \{ 洋室, 窓 \}$



全体のエラー傾向の分析2

- ・ エラーと周辺状況との関連性を分析

相関ルール分析における確信度と支持度の考え方を利用

正解ラベル A と、その周辺状況(隣接ノード) S

確信度 $P(B | A, S)$: A の S がある条件で $A \Rightarrow B$ が起こる確率

支持度 $P(A, S)$: A に対する S が起こる確率

※ S の全部分集合を考慮

ルールの一般性の尺度

全体のエラー傾向の分析結果2 誤認識

correct label ⇒ wrong label	{S}	conf
ダイニング⇒DK	{廊下, 洋室}	0.2759
	{洋室}	0.2759
	{収納, 廊下, 洋室}	0.2771
	{廊下}	0.2500
	{ }	0.2500
LD⇒LDK	{廊下}	0.1719
	{廊下, キッチン}	0.1719
	{キッチン}	0.1675
	{ }	0.1675
LDK⇒LD	{廊下, 洋室}	0.0769
	{ }	0.0682
LD⇒DK	{キッチン}	0.0262
	{ }	0.0262
LDK⇒リビング	{廊下, 洋室}	0.0385
	{洋室}	0.0333
	{ }	0.0227

correct label ⇒ wrong label	{S}	conf
ダイニング⇒LDK	{収納, 和室, 廊下, 洋室}	0.1212
	{収納, 和室, 洋室}	0.1212
	{収納, 廊下, 洋室}	0.1143
	{収納, 洋室}	0.1081
	{収納, 和室, 廊下}	0.0769
	{収納, 和室}	0.0769
	{収納, 廊下}	0.0714
	{廊下, 洋室}	0.0690
	{収納}	0.0690
	{洋室}	0.0690
	{廊下}	0.0690
	{ }	0.0417

- キッチンが正しく認識できていない
- 廊下や洋室がエラーを誘発

状況の違いによるエラー傾向の分析

- 間取り図の表示順序に特徴を持たせる
- 間取り図を見慣れている(1名)・見慣れていない(9名)
- タスクの前半部分(前半20件)・後半部分(後半20件)

エラーの傾向が異なるはず！

状況の違いによるエラー傾向の分析

- 間取り図の表示順序に特徴を持たせる

より似ている間取り図を連続して表示するグループ(5名)

より似ていない間取り図を連続して表示するグループ(5名)

状況の違いによるエラー傾向の分析結果

- 間取り図あたりのエラー平均回数と変換平均時間

グループ名	誤操作	誤認識	不認識	時間(秒)
類似グループ	1.1050	2.2800	5.0850	220.5
非類似グループ	2.0250	1.6550	5.8750	308.7
慣れグループ	0.8625	1.7375	3.8750	234.9
不慣れグループ	1.7406	2.0250	5.8813	271.9

誤操作 1.83倍
誤認識 0.72倍
不認識 1.16倍
時間 1.40倍

タスク中に表示される間取り図が大きく変わる

→ 誤操作が多くなる一方, 不認識はやや増加
間取り図を入念に見るため誤認識が減少

状況の違いによるエラー傾向の分析結果

- 間取り図あたりのエラー平均回数と変換平均時間

グループ名	誤操作	誤認識	不認識	時間(秒)
類似グループ	1.1050	2.2800	5.0850	220.5
非類似グループ	2.0250	1.6550	5.8750	308.7
慣れグループ	0.8625	1.7375	3.8750	234.9
不慣れグループ	1.7406	2.0250	5.8813	271.9

誤操作 2.02倍
誤認識 1.17倍
不認識 1.52倍
時間 1.16倍

間取り図を見慣れていない

誤認識・不認識ともに増加

特に構造を瞬時に理解することができず誤操作が大きく増加

状況の違いによるエラー傾向の分析結果

- 間取り図あたりのエラー平均回数と変換平均時間

グループ名	誤操作	誤認識	不認識	時間(秒)
前半グループ	1.8150	2.1400	5.5650	307.7
後半グループ	1.3150	1.7950	5.3950	222.1

誤操作 0.73倍
誤認識 0.84倍
不認識 0.96倍
時間 0.72倍

タスクに慣れる

→ 間取り図構造への理解が深まることにより、作業時間だけでなく、各エラー率も減少

間取り図グラフ作成における誤認識・不認識の分析 まとめ

- 誤認識・不認識の起こりやすい箇所とその周辺状況の関連性が明らかになった
- 間取り図の表示順序等の状況の違いとエラーの関係性に特徴的な傾向がみられた



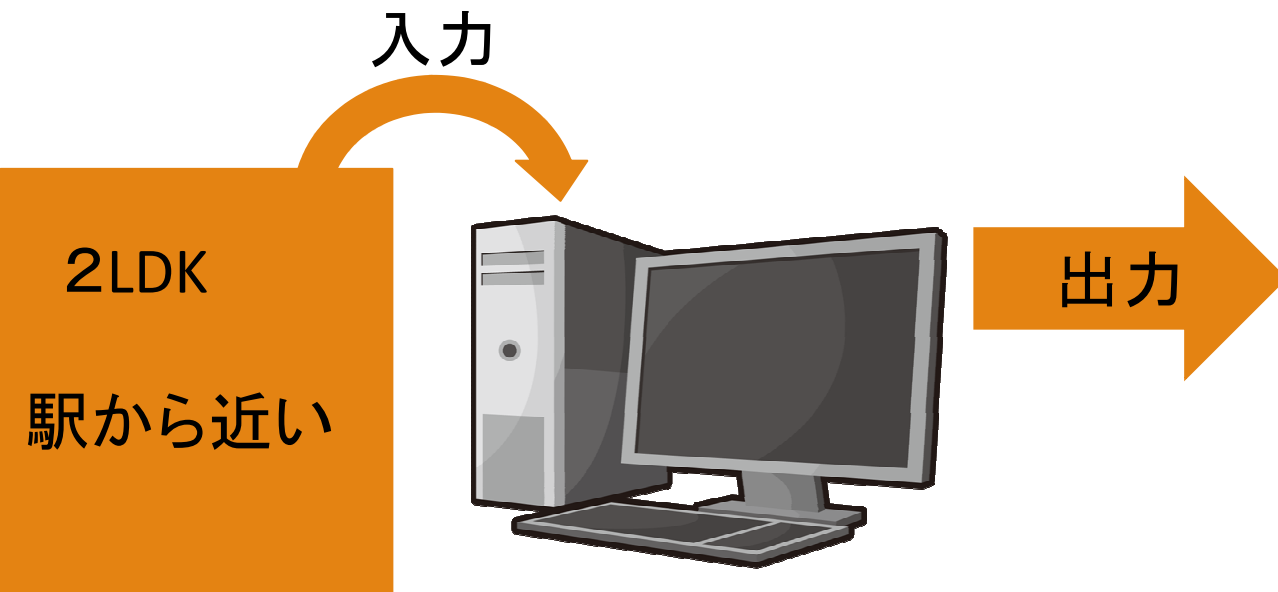
- 間取り図の正確な理解から、部屋探しにおける効率化の推進
- クラウドソーシングにおいて高品質なデータの獲得の可能性の向上

間取り図グラフ作成における誤認識・不認識の分析 今後の課題

- 実験時における制約や, ラベル属性のグループ化の強化
- 実験時のエラーと集中力との相関分析

間取り図画像のグラフ化による 検索支援に関する研究

グラフを用いたグラフの検索

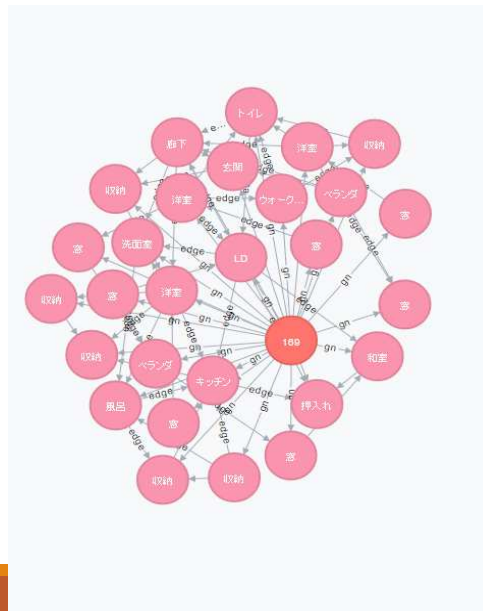


検索結果から間取り図を見比べるしかない。

グラフデータベース

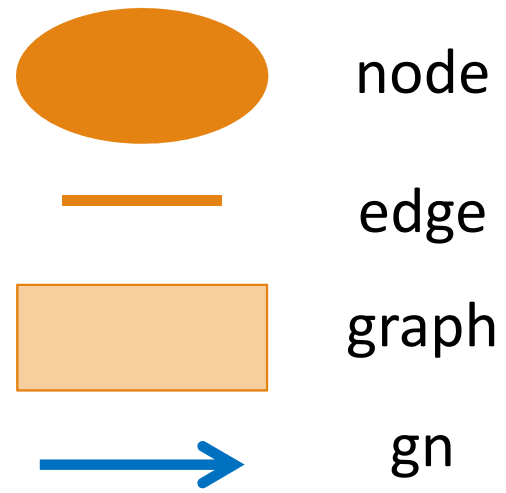
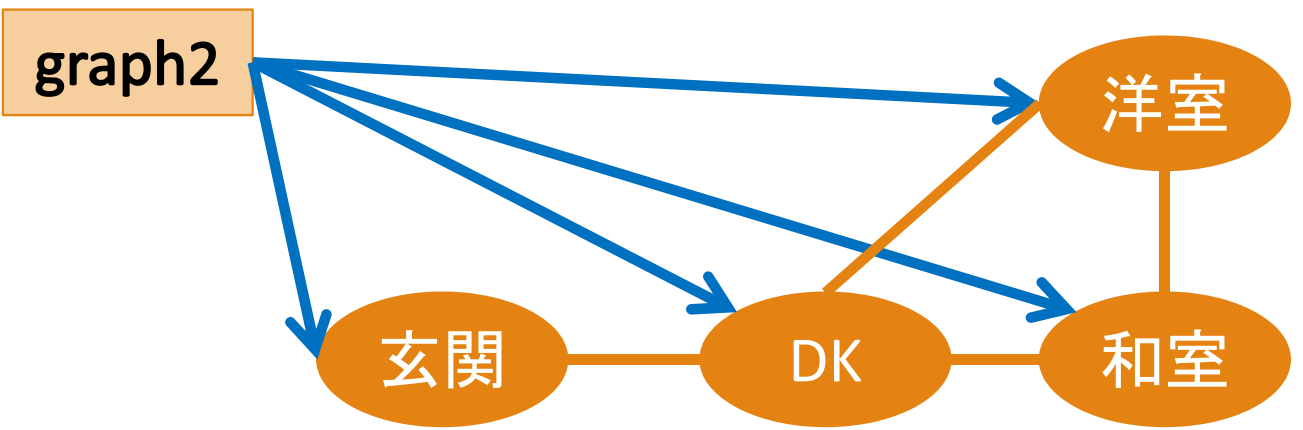
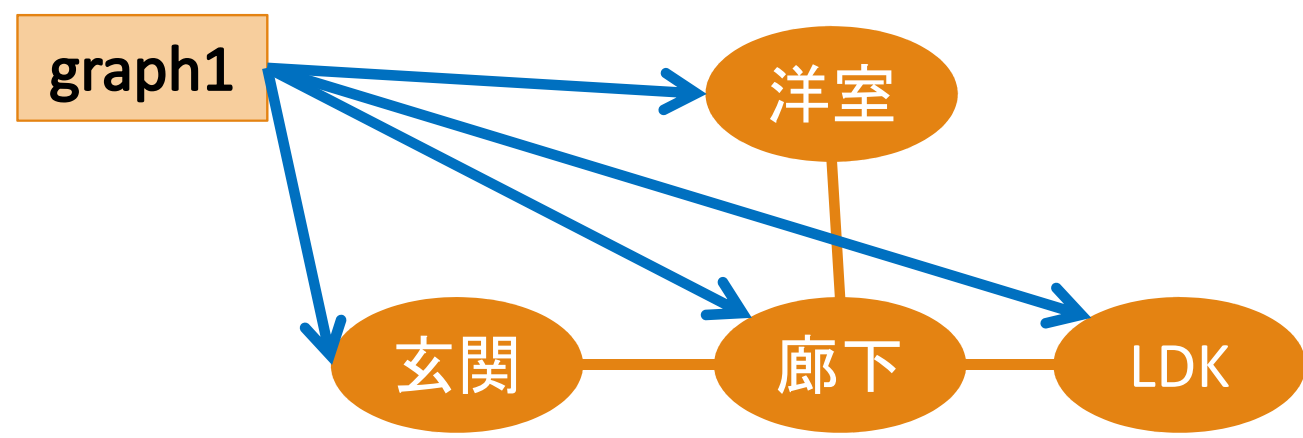
	グラフデータベース	関係データベース
データ形式	グラフ	表形式
実装	Neo4j	MySQL・Postgres
問い合わせ言語	Cypher	SQL

グラフを使うことで、
部屋の配置について検索できる。



```
+-----+-----+
| rooms | count |
+-----+-----+
| 3 | 35198 |
| 4 | 1297 |
| 5 | 72 |
| 6 | 25 |
| 21 | 1 |
| 23 | 1 |
| 34 | 1 |
+-----+-----+
```

Neo4jデータベース



Neo4jでは複数のグラフを入れることができない

graphノードを使って管理する

Cypher言語

グラフデータベースでの検索をしたい。

Neo4jはCypher言語を用いることで、簡単にグラフ検索できる。

Cypherの基本

MATCH 後に続く条件のグラフを検索する

WHERE 細かい検索の条件を決める

RETURN 検索結果を返す

SQLの基本

SELECT 抽出する属性

FROM テーブルを指定

WHERE 条件を指定

例

```
MATCH (g:graph) --> (n1:node)
```

```
WHERE n1.kind = "洋室"
```

```
RETURN distinct g.name
```



graph



洋室

これは、洋室がある間取りを全て取得している

検索時に求められる条件の例

部屋の繋がりに関して

- 客間とトイレは近いか
- 子供部屋、LD、玄関の順に繋がっている
- リビングに収納がある
- 風呂、トイレ、洗面室が一体でない

窓に関して

- 日当たり
- 風抜け

これらがCypherで書けるか確認する

その他

- 収納の数
- 柱や梁が出てないか

参考webサイト

間取り図の見方

<<http://www.housecom.jp/mikata/>>

一人暮らしの間取り公開。住み心地が変わる9つのポイント

<[http://onerom-navi.jp/a-single-life-](http://onerom-navi.jp/a-single-life-the-plan-of-a-house)

[the-plan-of-a-house](http://onerom-navi.jp/a-single-life-the-plan-of-a-house)[the-arrangement-of-the-rooms-481](http://onerom-navi.jp/a-single-life-the-plan-of-a-house)>

検索に用いるCypher言語

例: トイレと洗面室が近いか

```
MATCH (g:graph)-->(a:node)-[e1:edge*.. 数字]-(b:node)
WHERE a.kind = "トイレ" and b.kind = "洗面室"
RETURN distinct(g.name)
```

二つの部屋が近い間取りを探す。

検索に用いるCypher言語

玄関から子供部屋に行くのにLDを通る

```
MATCH (g:graph),(g)-->(a:node),(g)-->(b:node),  
  p = shortestPath((a)-[:edge*]- (b))  
WHERE a.kind="玄関" and b.kind="洋室"  
  and all(r IN relationships(p)  
    WHERE r.kind="ドア" or r.kind="無し" or r.kind="引き戸")  
  and any(r2 in nodes(p) WHERE r2.kind="LD")  
RETURN distinct g.name
```

玄関から子供部屋(洋室)に移動する際に、LDを通る。

検索に用いるCypher言語

南側に窓がある部屋がある

```
MATCH(g:graph)-->(n:node)--(n2:node)
WHERE n.kind = "部屋A"
and n2.kind = "窓"
and (atan2(n.x-n2.x ,n.y-n2.y) - g.direction - PI()/2 +4*PI())
%(2*PI()) < PI()/8
RETURN distinct g.name
```

窓が南向きにある部屋を探せる。南以外の向きも検索可能。

Cypher言語に変換できないもの

二つの部屋の経路上にあるドアの数

二つの経路上にはドア以外のエッジも存在する。
これらを仕分けることが困難であるから。

柱や梁によるでっぱりの有無

柱や梁のように部屋の形に関わる情報が、グラフにする過程で失われているから。

風通し

窓が部屋のどの面にあるのかという情報が、グラフにする過程で失われているから。

Cypher言語以外も組み合わせる必要がある。

まとめ

- 検索で間取り図の部屋同士の繋がりについて検索することができた。
- 繋がり以外のグラフでは表せない情報についての条件を検索できなかった。
- 検索時に求められる条件の半分以上はCypherで表せた。

グラフマイニングを用いた 間取り図情報に基づく賃料分析

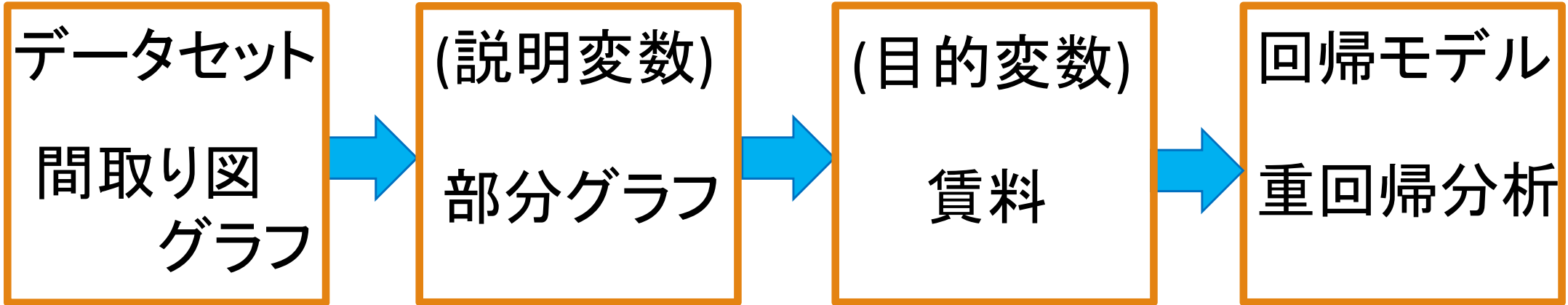
背景 & 目的

- 物件の部屋面積や、部屋階数が賃料に影響を与えているのは明らか

- どのような部屋配置がどの程度賃料に影響を与えているのかを分析

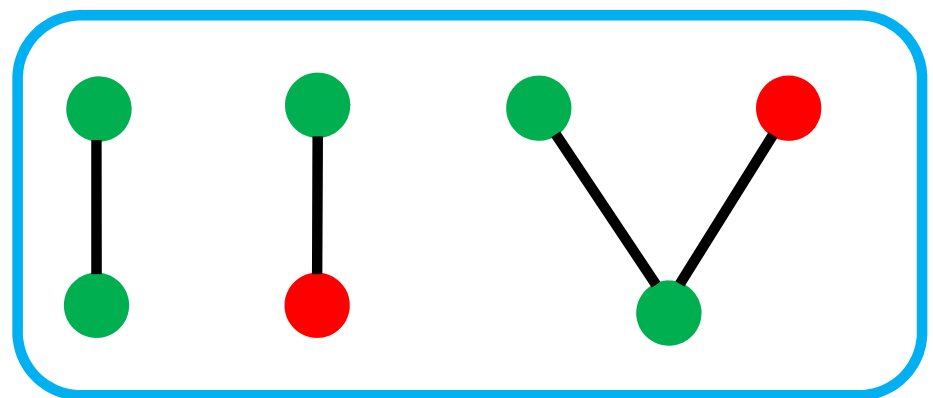
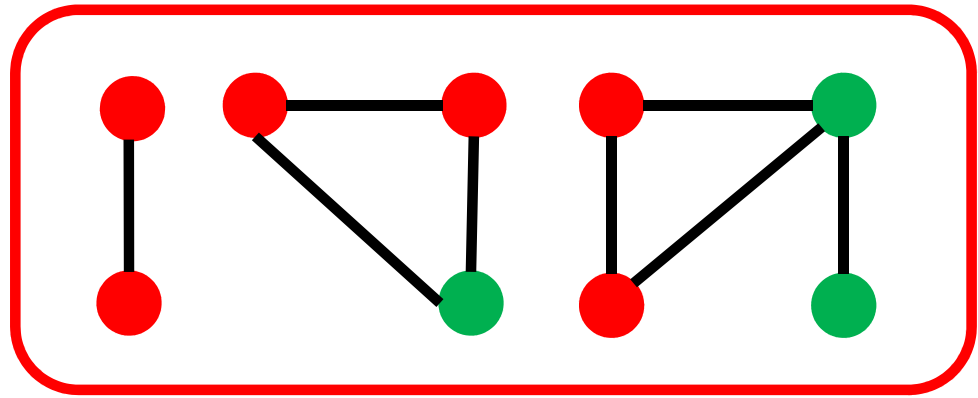
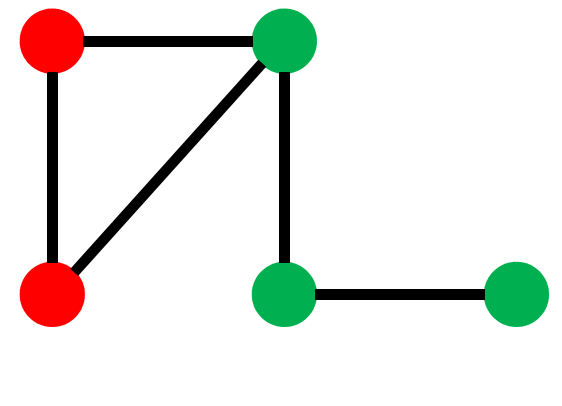
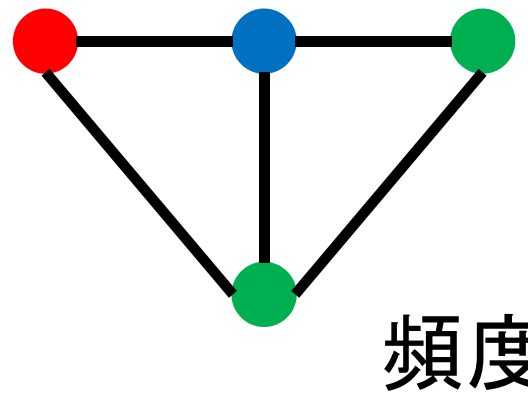
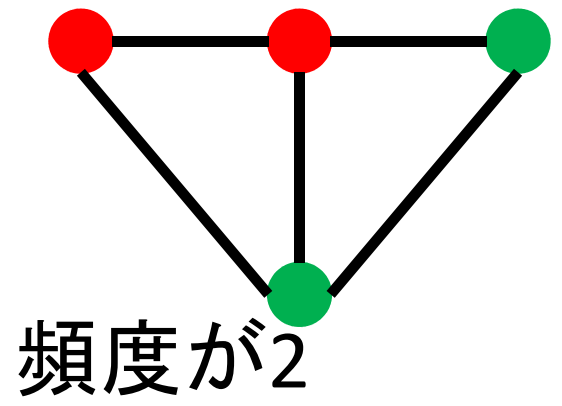


分析の流れ



- 部屋面積
- 部屋階数
- …
- 頻出部分グラフ
- FreePattern
- サイズが2以上
- 頻度が中程度
- …
- 賃料と相場
の偏差
- 賃料と相場
の比
- …
- 回帰木
- モデル木
- SVR

頻出部分グラフ

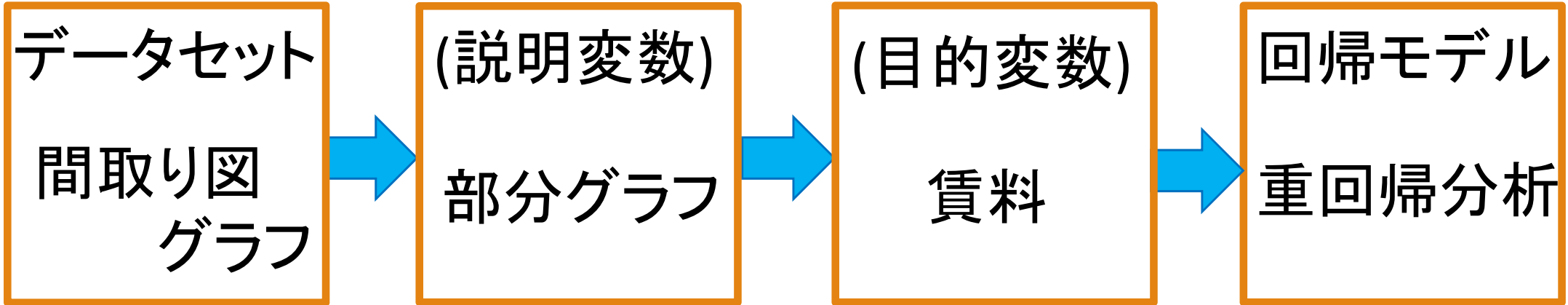


FreePattern

説明変数の種類


1. グラフを用いない
2. 頻度が中程度
3. サイズが2以上
4. サイズが3以上
5. サイズが4以上
6. FreePattern&サイズが2以上
7. FreePattern&サイズが3以上
8. FreePattern&頻度が低程度
9. FreePattern&頻度が中程度
10. FreePattern&頻度が中程度

分析の流れ



- 部屋面積
 - 部屋階数
 - …
- 頻出部分グラフ
 - FreePattern
 - サイズが2以上
 - 頻度が中程度
 - …
- 賃料と相場
の偏差
 - 賃料と相場
の比
 - …
- 回帰木
 - モデル木
 - SVR

目的変数の種類

- A) 賃料 賃料の平均 : 238305円
 - B) 賃料 + 説明変数に相場
 - C) (賃料 - 相場) 賃料と相場の偏差
 - D) (賃料/相場) 賃料と相場の比
- 

分析の流れ



データセット
間取り図
グラフ

(説明変数)
部分グラフ

(目的変数)
賃料

★
回帰モデル
重回帰分析

- 部屋面積
- 部屋階数
- ⋮

- 頻出部分グラフ
- FreePattern
- サイズが2以上
- 頻度が中程度
- ⋮

- 賃料と相場
の偏差
- 賃料と相場
の比
- ⋮

- 回帰木
- モデル木
- SVR

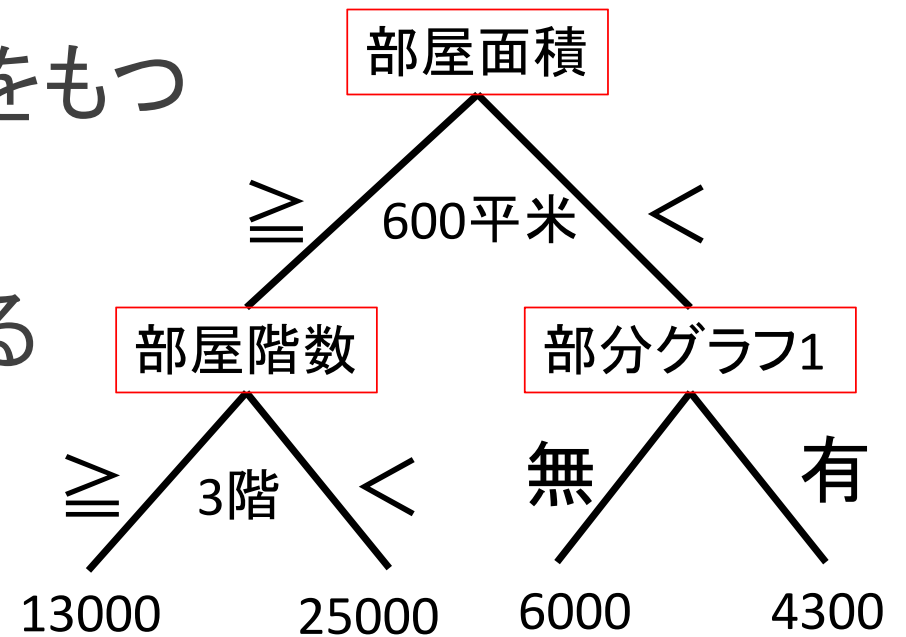
重回帰分析 (線形回帰)

$$Y = a_1 * X_1 + a_2 * X_2 + \dots + b$$

$$\begin{aligned} \text{賃料} = & 1000 * \text{部屋面積} + 300 * \text{部分グラフ1を含む} \\ & + \dots + 10000 \end{aligned}$$

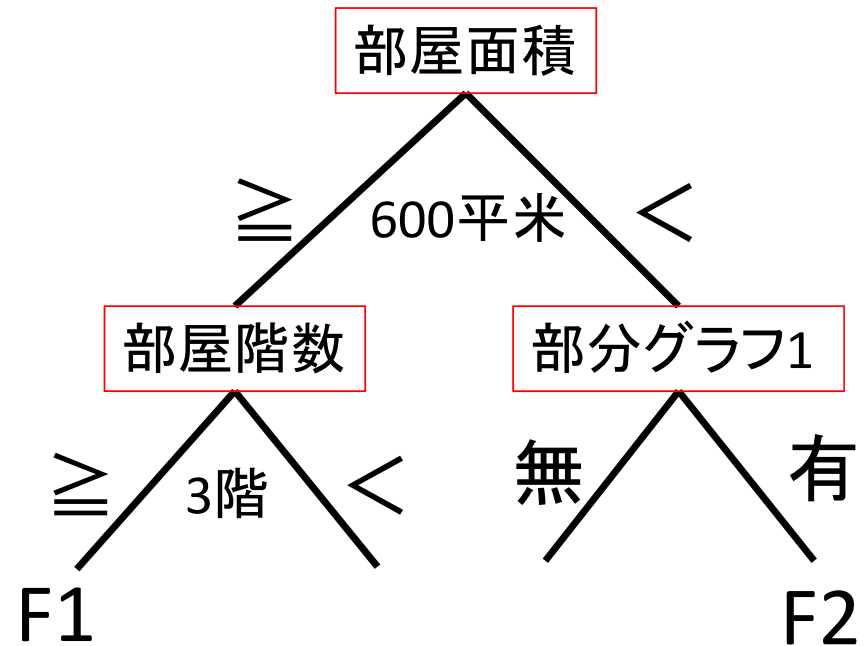
回帰木 (非線形回帰)

- 木構造であり、葉に予測値 (賃料) をもつ
- CARTを用いて2分回帰木を構築する



モデル木 (場合分けをした線形回帰)

- 木構造であり、
葉に線形回帰式をもつ



F1

F2 賃料 = 1000 * 部屋面積 + 500 * 部屋階数 + ... + 10000

賃料 = 800 * 部屋面積 + 30 * 部分グラフ1を含む + ... + 4500

目的変数に着目した精度の比較

- 精度：平均絶対誤差（実測値と予測値の差の平均）
低いほど精度が高い

	平均	最小
賃料	53617.27	35012
賃料 + 説明変数に相場	53960.32	35570
(家賃 - 相場) 賃料と相場の偏差	50668.45	31098
(家賃/相場) 賃料と相場の比	42168.39	29896

説明変数に着目した精度の比較

	平均	最小
グラフ無し	44031.55	30179
頻度が中程度	49740.78	30108
サイズが2以上	49567.03	29896
サイズが3以上	48816.13	30292
サイズが4以上	47181.91	31474
FreePattern&サイズが2以上	56076.65	31680
FreePattern&サイズが3以上	55007.60	32609
FreePattern&頻度が低程度	62148.86	31618
FreePattern&頻度が中程度	44106.37	31135
FreePattern&頻度が高程度	44375.81	31490

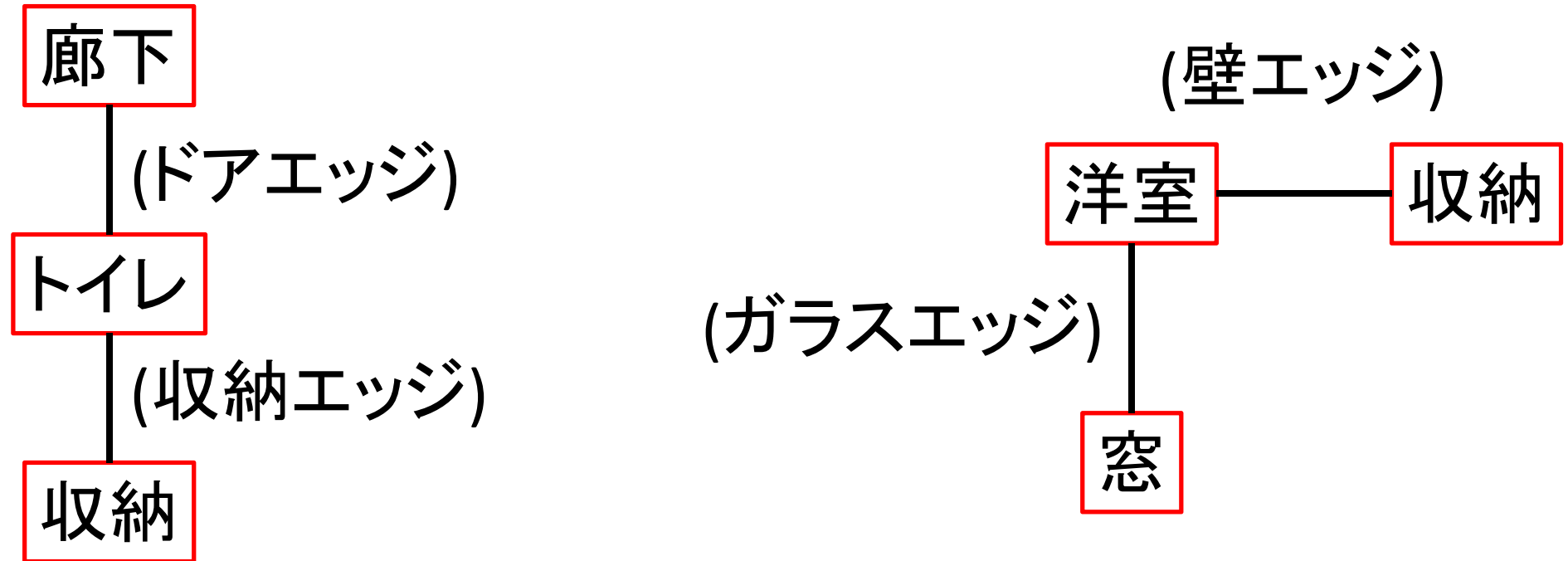
回帰モデルに着目した精度の比較

	平均	最小
回帰木	37366.22	29895.58
モデル木	34883.71	30604.84
SVR	62664.86	41598.47
重回帰分析	65506.28	41728.49

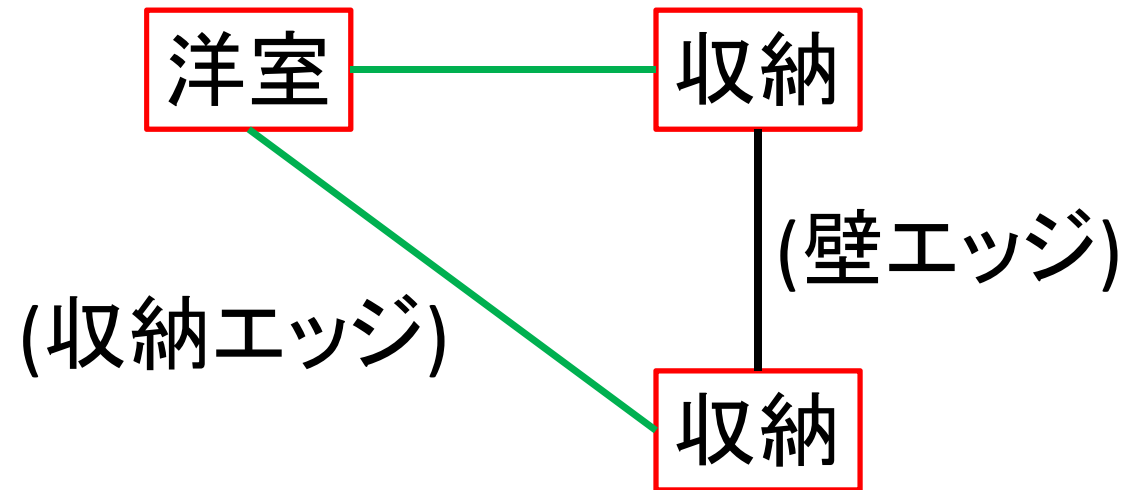
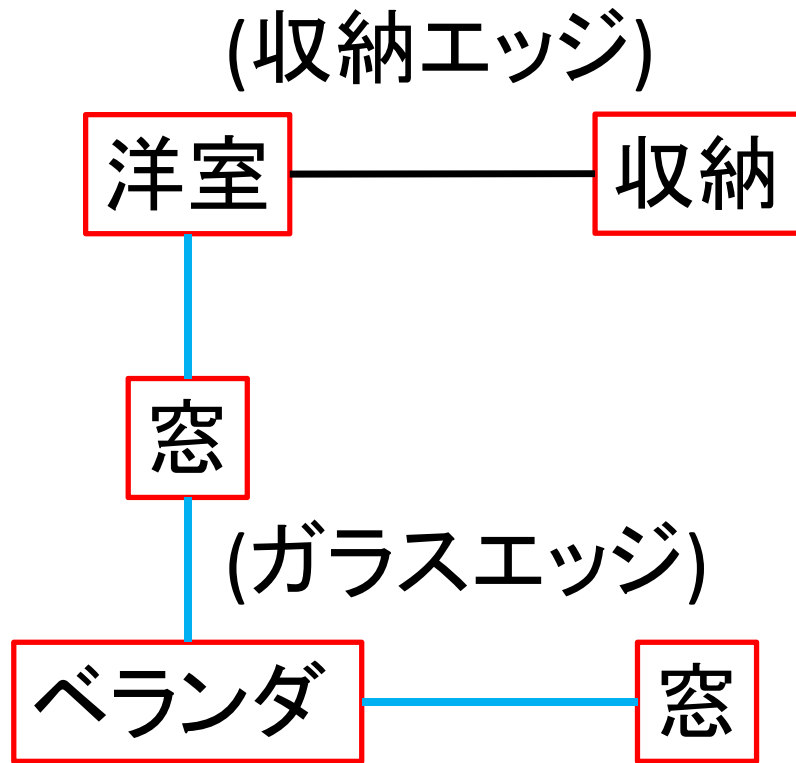
目的変数・説明変数・モデル回帰 全てを考慮した最も精度が高い組み合わせ

- 目的変数 : (賃料/相場) 賃料と相場の比
- 説明変数 : サイズが2以上
- 回帰モデル : 回帰木

回帰木による賃料に影響を与える部屋配置



モデル木による賃料に影響を与える部屋配置



賃料分析に関して まとめと今後の課題

- 既存手法と比較し、本手法が優れた
- 賃料に影響を与える部屋配置とその程度を確認
- 数ある部屋配置の内、ある部屋配置と賃料との因果効果を、傾向スコア分析で調べる