

# 基礎理論ゼミ -交通ネットワーク-

## Rで実装 最短経路探索法

2016/6/6

福田研究室 修士2年 篠原丈実



# 0.目次

1. イントロ
2. ダイクストラ法(Liuくん担当のため割愛)
3. ラベル修正法

最短経路探索法の理論→次回の**第8章**

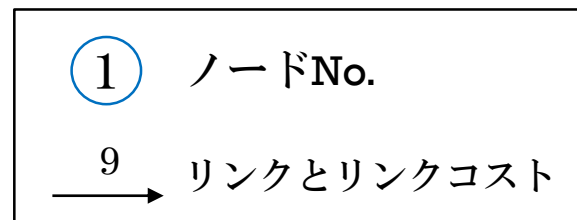
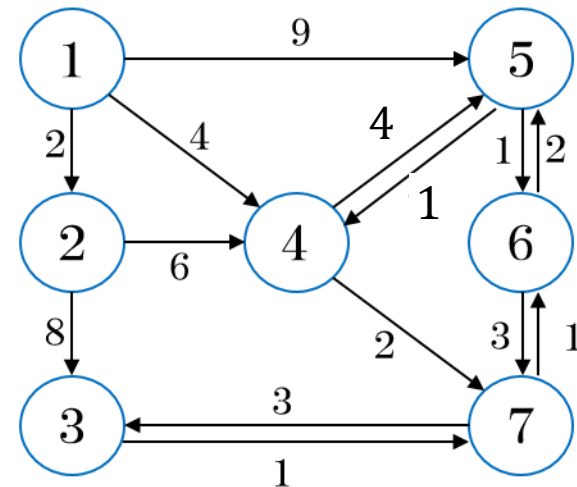
# 1. イントロ

★最短経路探索法・・・利用者均衡配分の計算アルゴリズム

1つの起点からすべての終点までの最短経路を1回の計算で同時に導出

右図のようなネットワークの最短経路探索を以下のアルゴリズムで行う

- ✓ダイクストラ法
- ✓ラベル修正法

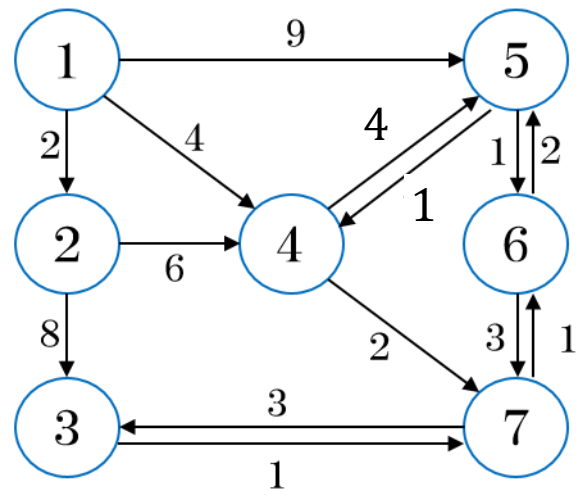


# 1. イントロ

## ☆Rでネットワークの作成

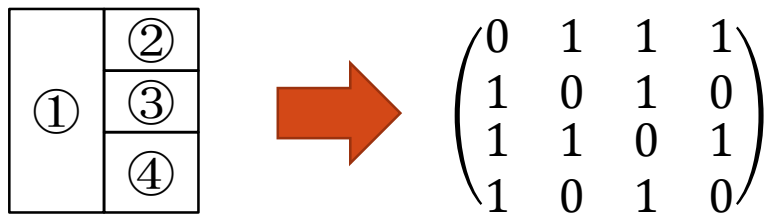
右図のようなネットワークをRで作図

✓ データ...”05172.csv”  
「隣接行列」の形式

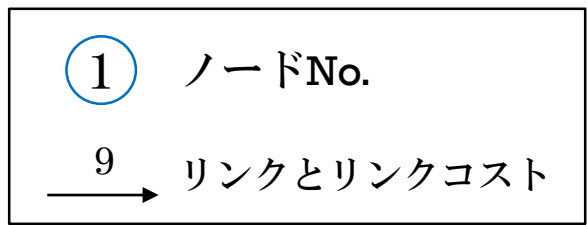


※隣接行列とは...

隣接関係(0 or 1)を行列で表示したもの



今回は隣接関係をリンクコストで重み付け  
=行列の要素がリンクコストに



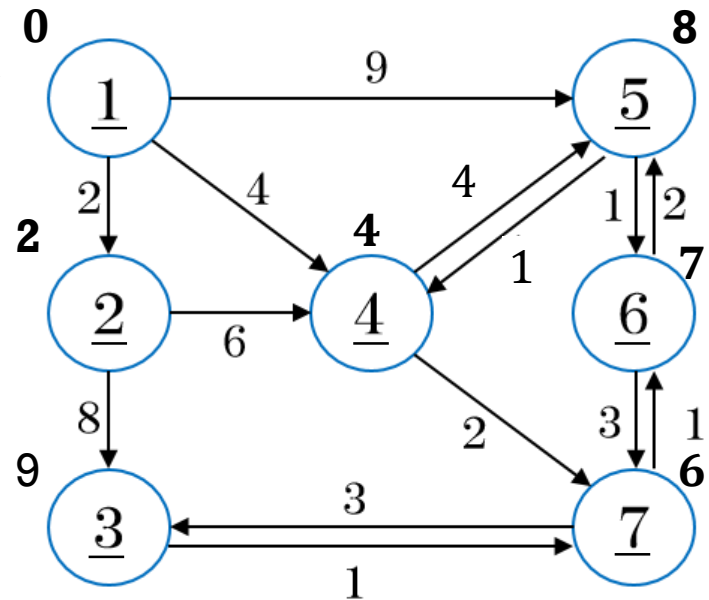
✓ プログラム...”ネットワーク作成.R”

## 2.ダイクストラ法

### ☆Rでダイクストラ法

右図のネットワークに関して最短経路を探索

- プログラム…“ダイクストラ法.R”



(参考)<http://tjo.hatenablog.com/entry/2015/11/17/190000>

# 3. ラベル修正法

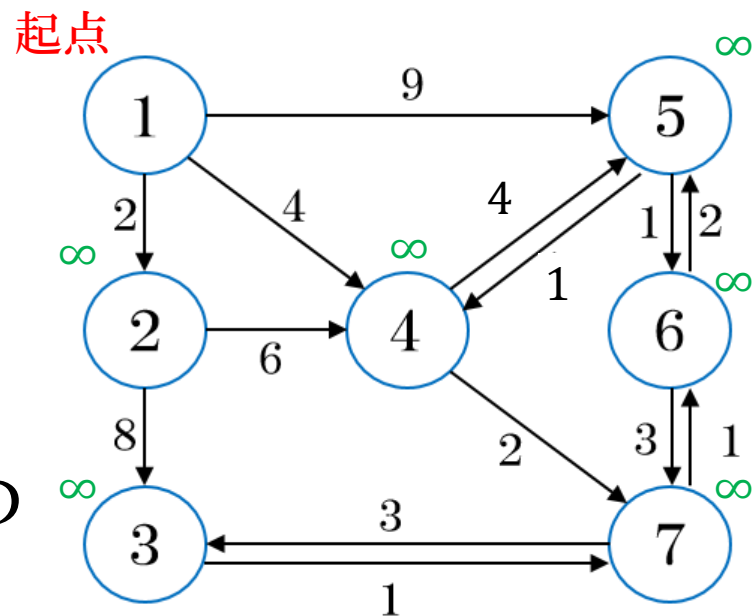
## ★ラベル修正法

※ダイクストラ法...

最小の交通費用を要するノードを探索

ラベル修正法  
では...

起点ノードとリンクで結ばれる全ての  
ノードについて費用を更新



はじめに起点をA{ノードリスト}に追加

★現時点では... A{1}

# 3. ラベル修正法

## ★ラベル修正法

① 起点(ノード1)における最小費用 ( $c_1 = 0$ )とし、集合Aの先頭ノードを取り出し、集合Aから削除

★現時点では... A = ∅

② ①のノードiから出るすべてのリンクの終点ノード{m}において

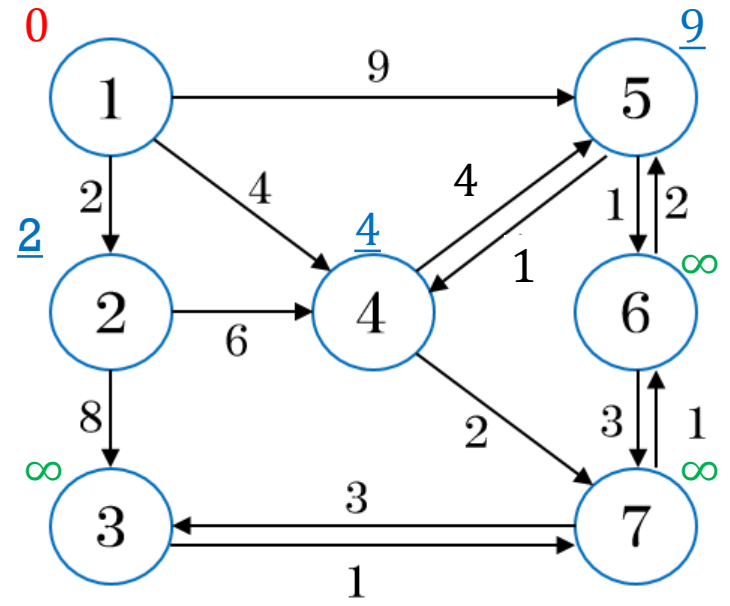
$$c_m > c_i + t_{im}$$

※ $t_{im}$ ...ノードi,ノードm間のリンクにおける費用

であれば  $c_m = c_i + t_{im}$  に更新し、集合Aに含まれていなければmを集合Aの最後に追加

★現時点では... A{2,4,5}

③更新された集合A に関して、①に戻る



赤数字... Aの先頭ノードの最小費用

青数字... 対象の終点ノードの最小費用  
(青数字は  $c_m$  が更新されたノード = 集合Aに追加)

# 3. ラベル修正法

## ★ラベル修正法

① 集合Aの先頭ノード2を削除

★現時点では・・・A = {4,5}

② ①のノード*i*から出るすべてのリンクの終点ノード{*m*}において

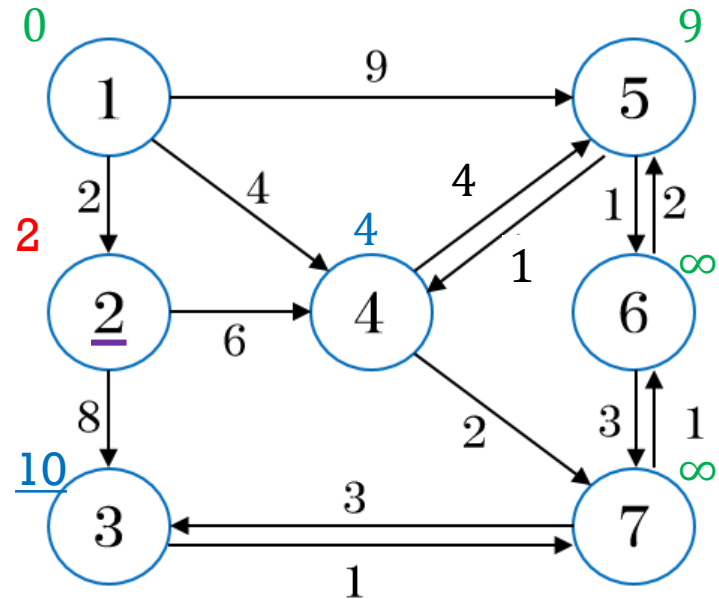
$$c_m > c_i + t_{im}$$

※ $t_{im}$ ・・・ノード*i*,ノード*m*間のリンクにおける費用

であれば $c_m = c_i + t_{im}$ に更新し、集合Aに含まれていなければ*m*を集合Aの最後に追加

★現時点では・・・A{4,5,3}

③更新された集合A に関して、①に戻る





# 3. ラベル修正法

## ★ラベル修正法

① 集合Aの先頭ノード4を削除

★現時点では・・・A = {5,3}

② ①のノード*i*から出るすべてのリンクの終点ノード{*m*}において

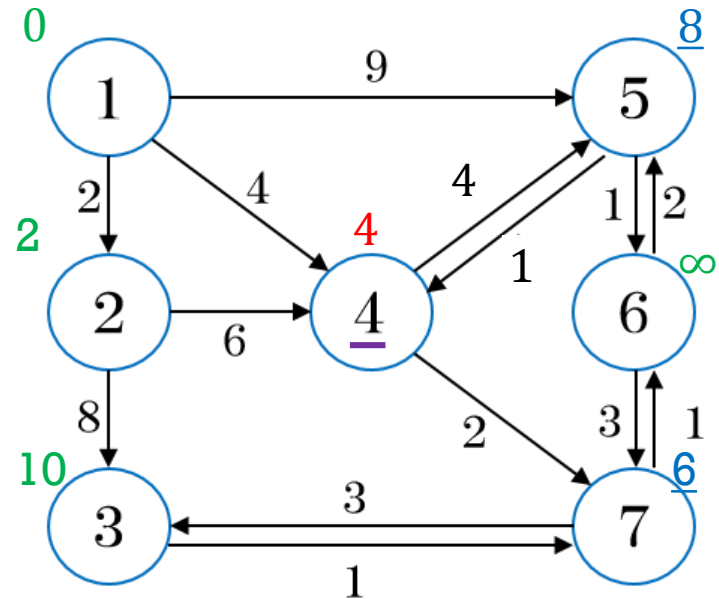
$$c_m > c_i + t_{im}$$

※ $t_{im}$ ・・・ノード*i*,ノード*m*間のリンクにおける費用

であれば $c_m = c_i + t_{im}$ に更新し、集合Aに含まれていなければ*m*を集合Aの最後に追加

★現時点では・・・A{5,3,7}

③更新された集合A に関して、①に戻る



※ポイント

・ノード5の最小費用も更新されたが既にAに属するため追加なし

# 3. ラベル修正法

## ★ラベル修正法

① 集合Aの先頭ノード5を削除

★現時点では・・・A = {3,7}

② ①のノード*i*から出るすべてのリンクの終点ノード{*m*}において

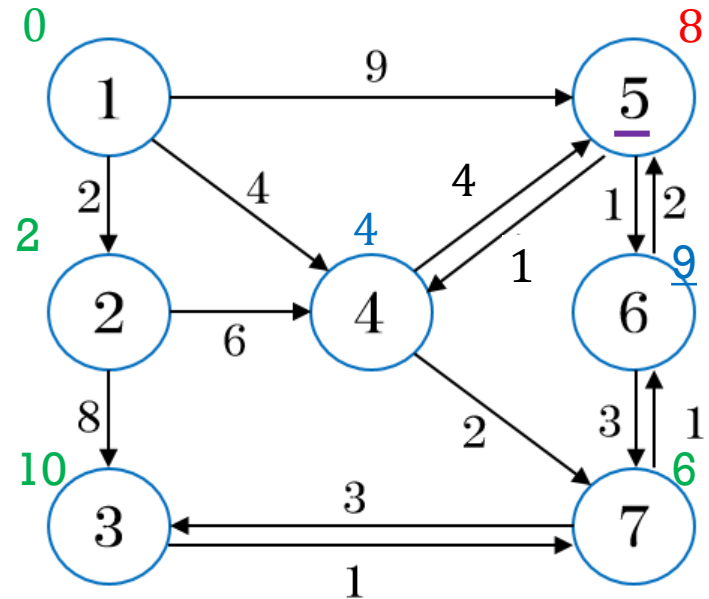
$$c_m > c_i + t_{im}$$

※ $t_{im}$ ・・・ノード*i*,ノード*m*間のリンクにおける費用

であれば $c_m = c_i + t_{im}$ に更新し、集合Aに含まれていなければ*m*を集合Aの最後に追加

★現時点では・・・A{3,7,6}

③更新された集合A に関して、①に戻る



# 3. ラベル修正法

## ★ラベル修正法

① 集合Aの先頭ノード3を削除

★現時点では・・・A = {7,6}

② ①のノード*i*から出るすべてのリンクの終点ノード{*m*}において

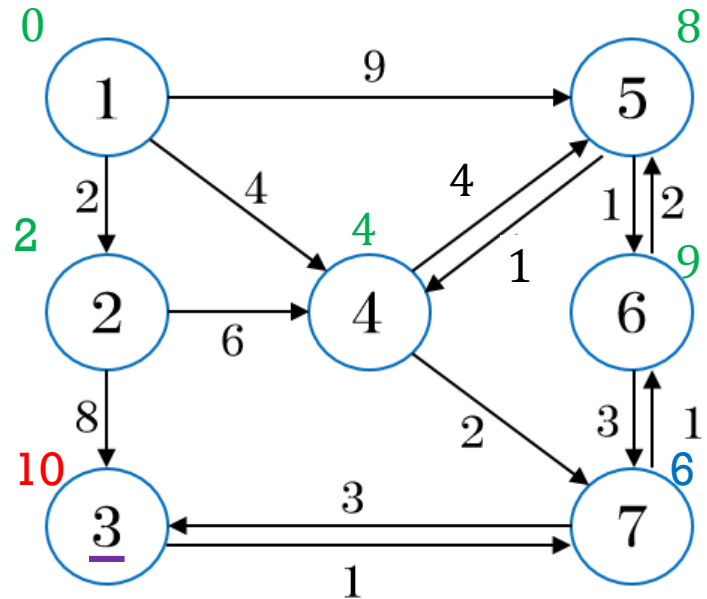
$$c_m > c_i + t_{im}$$

※ $t_{im}$ ・・・ノード*i*,ノード*m*間のリンクにおける費用

であれば $c_m = c_i + t_{im}$ に更新し、集合Aに含まれていなければ*m*を集合Aの最後に追加

★現時点では・・・A{7,6}

③更新された集合A に関して、①に戻る



# 3. ラベル修正法

## ★ラベル修正法

① 集合Aの先頭ノード7を削除

★現時点では・・・A = {6}

② ①のノード*i*から出るすべてのリンクの終点ノード{*m*}において

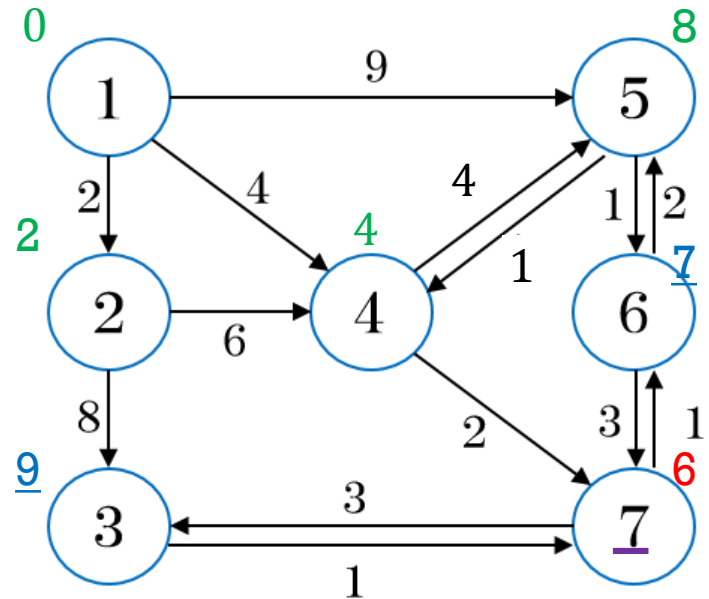
$$c_m > c_i + t_{im}$$

※ $t_{im}$ ・・・ノード*i*,ノード*m*間のリンクにおける費用

であれば $c_m = c_i + t_{im}$ に更新し、集合Aに含まれていなければ*m*を集合Aの最後に追加

★現時点では・・・A{6,3}

③更新された集合A に関して、①に戻る



※ポイント

・ラベル修正法では集合Aから一度除去されたノード(今回は3)が再登場する可能性あり

# 3. ラベル修正法

## ★ラベル修正法

① 集合Aの先頭ノード6を削除

★現時点では・・・A = {3}

② ①のノード*i*から出るすべてのリンクの終点ノード{*m*}において

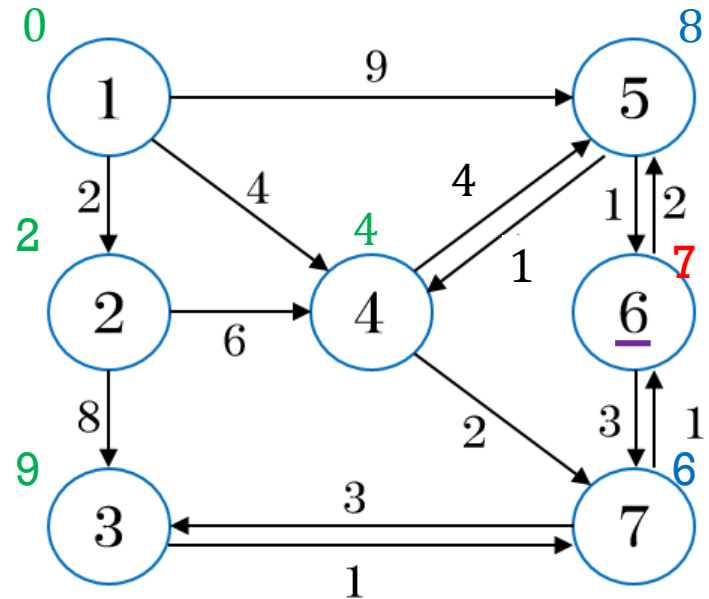
$$c_m > c_i + t_{im}$$

※ $t_{im}$ ・・・ノード*i*,ノード*m*間のリンクにおける費用

であれば $c_m = c_i + t_{im}$ に更新し、集合Aに含まれていなければ*m*を集合Aの最後に追加

★現時点では・・・A = {3}

④更新された集合Aが空集合になれば終了



# 3. ラベル修正法

## ★ラベル修正法

① 集合Aの先頭ノード3を削除

★現時点では... A = ∅

② ①のノードiから出るすべてのリンクの終点ノード{m}において

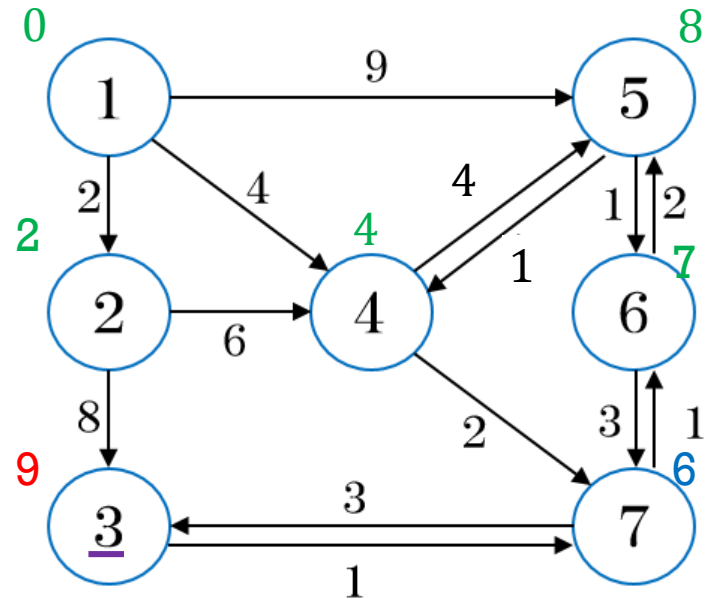
$$c_m > c_i + t_{im}$$

※ $t_{im}$ ...ノードi,ノードm間のリンクにおける費用

であれば $c_m = c_i + t_{im}$ に更新し、集合Aに含まれていなければmを集合Aの最後に追加

★現時点では... A = ∅

④更新された集合Aが空集合になれば終了



# 3. ラベル修正法

## ☆Rでラベル修正法

右図のネットワークに関して最短経路を探索

- プログラム…“ラベル修正法.R”

