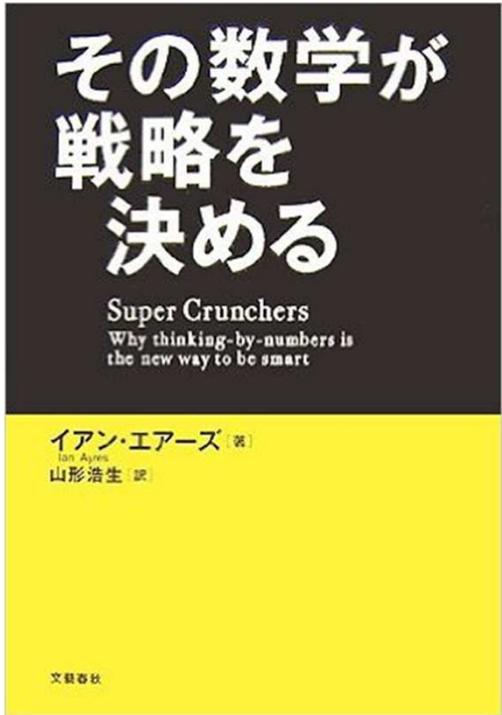


『アッセンフェルターのワイン方程式』を検証する



$$\begin{aligned}
 \text{ワインの質} &= 12.145 \\
 &+ 0.00117 \times \text{冬の降雨} \\
 &+ 0.0614 \times \text{育成期平均気温} \\
 &- 0.00386 \times \text{収穫期降雨}
 \end{aligned}$$

ワインの質 = $12.145 + 0.00117 \times \text{冬の降雨} + 0.0614 \times \text{育成期平均気温} - 0.00386 \times \text{収穫期降雨}$

ご覧の通り。どの年でも気象情報をこの方程式に入れば、アッセンフェルターはそのヴァンテージのおおまかな品質を予測できる。もうちょっと複雑な式を使うと、一〇〇以上のシャトーでヴァンテージ品質についてもっと厳密な品質が予測できる。



BlueMixでやってみた



The screenshot shows the IBM Bluemix console interface. At the top, it says "IBM Bluemix Ready? Try the new Bluemix | New! Try OpenWhisk". Below that, there's a navigation bar with a back arrow, the text "'Second Trial' のダッ...", and a "概要" (Overview) tab. The main content area shows a deployment card for "HandsOn001" with a rocket icon and the URL "HandsOn001.mybluemix.net". Below this, there are configuration options: "SDK FOR NODE.JS™" with a ".js" icon, "インスタンス:" (Instances) set to 1, "メモリー割り当て量:" (Memory allocation) set to 256 MB, and "使用可能メモリー:" (Available memory) set to 1.750 GB. There are "保存" (Save) and "リセット" (Reset) buttons. Below the configuration are two buttons: "+ サービスまたは API の追加" (Add service or API) and "+ サービスまたは API のバインド" (Bind service or API). At the bottom, there are two service cards: "dashDB" (dashDB-xr Entry) and "Predictive Analytics" (Predictive Analyti... Free). Each service card has a "資格情報の表示" (Show credentials) button and a "資料" (Documentation) button.

今回ご紹介するのはこの2機能

“R-Studio”がそのまま使える

“BlueMix初心者”
でも問題なし！

The screenshot displays the RStudio interface within a browser window. The console shows the execution of a linear regression model and its summary statistics.

```
> result <- lm(formula = LPRICE2 ~ WRAIN + DEGREES + HRAIN + TIME_SV, data = df01)
> summary(result)
```

Call:
lm(formula = LPRICE2 ~ WRAIN + DEGREES + HRAIN + TIME_SV, data = df01)

Residuals:

Min	1Q	Median	3Q	Max
-0.46027	-0.23864	0.01347	0.18600	0.53446

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-1.215e+01	1.688e+00	-7.195	3.28e-07 ***
WRAIN	1.167e-03	4.820e-04	2.421	0.02420 *
DEGREES	6.164e-01	9.518e-02	6.476	1.63e-06 ***
HRAIN	-3.861e-03	8.075e-04	-4.781	8.97e-05 ***
TIME_SV	2.385e-02	7.167e-03	3.328	0.00306 **

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2865 on 22 degrees of freedom
Multiple R-squared: 0.8275, Adjusted R-squared: 0.7962
F-statistic: 26.39 on 4 and 22 DF, p-value: 4.058e-08

The right-hand side of the interface shows a 'Residuals vs Fitted' plot. The y-axis is labeled 'Residuals' and ranges from -0.4 to 0.4. The x-axis is labeled 'Fitted values' and ranges from -2.5 to -0.5. A red regression line is plotted through the data points. Several points are labeled with their observation numbers: 60, 24, and 20.

結果をよく見ると...

Call:

lm(formula = LPRICE2 ~ WRAIN + DEGREES + HRain + TIME_SV, data = df01)

Residuals:

Min	1Q	Median	3Q	Max
-0.46027	-0.23864	0.01347	0.18600	0.53446

よく見ると
マイナス

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-1.215e+01	1.688e+00	-7.195	3.28e-07 ***
WRain	1.167e-03	4.820e-04	2.421	0.02420 *
DEGREES	6.164e-01	9.518e-02	6.476	1.63e-06 ***
HRain	-3.861e-03	8.075e-04	-4.781	8.97e-05 ***
TIME_SV	2.385e-02	7.167e-03	3.328	0.00306 **

ん、さっきの式に
これが
無いぞ?!

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2865 on

ワインの年齢 × 0.02358

Multiple R-squared: 0.8275, Adjusted R-squared: 0.7962

F-statistic: 26.39 on 4 and 22 DF, p-value: 4.058e-08

『ビールとおむつ』を 探し出す

『ビールとおむつ』を探り出す



Market Basket Analysys , サンプル通り実行

nakanita@c

The screenshot shows the IBM dashDB interface. On the left is a dark sidebar menu with options: Home, Tables, Load, Run SQL, Analytics (circled in red), Monitor, Settings, Connect, and Help. The main content area has a header 'Run R scripts to analyze, manipulate, and visualize your data' and a sub-header 'Create a new script, import a script, or use a sample script. Learn more'. Below this is a 'Sample Projects' section with two expandable categories: 'In-Application Analytics' and 'In-Database Analytics'. Under 'In-Database Analytics', 'Market Basket Analysis' is selected and circled in red. To the right of the project list, the 'Script Name: Market Basket Analysis' is displayed, along with a description: 'This sample shows the use of association rule mining to often purchased in the same transaction.' Below the description are tabs for 'Script', 'Console Output', and 'Plots'. The 'Script' tab is active, showing a code editor with R code. At the top of the script editor are buttons for 'Submit', 'Add a Data Frame...', and 'Save'.

IBM dashDB™

Run R scripts to analyze, manipulate, and visualize your data
Create a new script, import a script, or use a sample script. [Learn more](#)

Script Name: Market Basket Analysis

Description: This sample shows the use of association rule mining to often purchased in the same transaction.

Click **Submit** to generate a plot. Errors, warnings, or messages are

Script Console Output Plots

Submit Add a Data Frame... Save

```
library(arulesViz)
library(ibmdbR)

con <- idaConnect("BLUDB",",")
idalnit(con)

#The table ORDER_DETAILS contains transactions from a fictitious retailer
#We are only interested in two columns, ORDER_NUMBER, the ID of the tra
#the id of the product purchased in this transaction
orderDetails <- ida.data.frame("GOSALES.ORDER_DETAILS")[,c("PRODUCT"
```

```
#In this sample we do not want to find association rules among products but
#product types. Therefore we merge the order details with the table PRODU
#contains among others a mapping from PRODUCT_NUMBER to PRODU
products <- ida.data.frame("GOSALES.PRODUCT")[,c("PRODUCT_NUMBEF
```

『ビルとおむつ』はを探り出す

データ読込は
Drag&Dropで

kanita@gmail

The screenshot shows the IBM dashDB interface. On the left is a dark sidebar menu with items: Home, Tables, Load (circled in red), Run SQL, Analytics, Monitor, Settings, Connect, and Help. The main content area is titled 'Load from desktop' and contains the following elements:

- Instruction: 'Load one time from an Excel file or a delimited text file, such as a comma-separated value (CSV) file. [Learn more](#)'
- Progress steps: 1. Specify source file (highlighted), 2. Choose the target, 3. Select a table, 4. Load complete
- Transfer mechanism: 'Standard' (selected), 'Aspera (BETA)', and 'Install Aspera Connect' with a help icon.
- Supported file types: 'Supported file types: CSV, XLS, XLSX'
- A dashed box representing a drop zone with a hand icon and the text 'Drag a file here or Choose a file'.

Annotations include a blue curved arrow pointing from a 'CSV' file icon to the drop zone, and a red speech bubble pointing to the drop zone with the text '対話式に DBテーブルができる'.

対話式に
DBテーブルが
できる

『ビールとおむつ』を探り出す

商品同士の 関連をグラフ化

The screenshot shows an RStudio session with the following code and output:

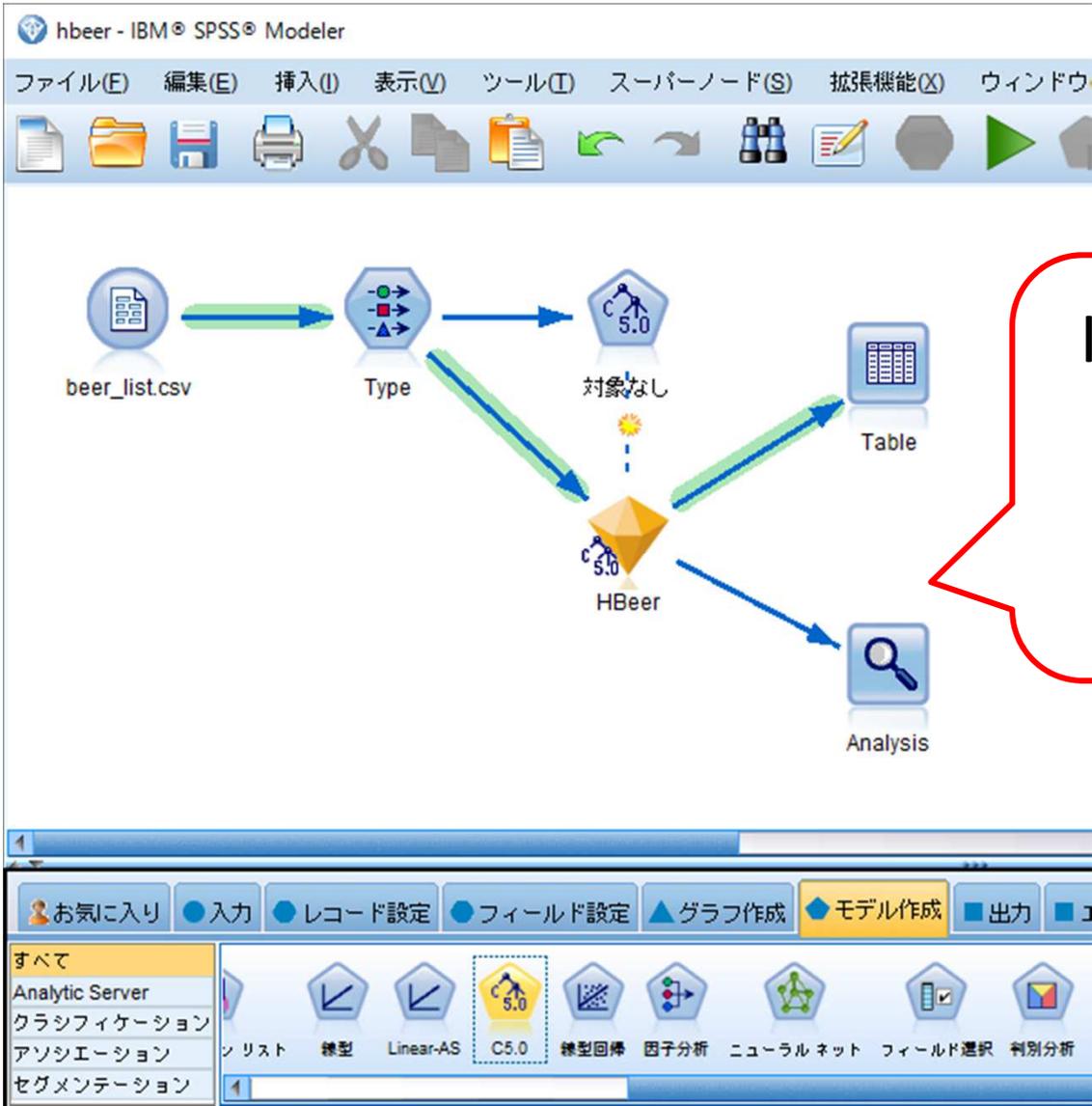
```
> library(ibmdbR)
> orderDetails <- ida.data.frame('DASH105336.ORDER_DETAILS')[,
c('PRODUCT_NUMBER', 'ORDER_NUMBER')]
> head(orderDetails)
  PRODUCT_NUMBER ORDER_NUMBER
1          1603           1383
2          44101           1383
3          44101           1383
4           2232           1384
5          44101           1384
6          43007           1384
>
> products <- ida.data.frame('DASH105336.PRODUCT')[,c('PRODUCT
NUMBER', 'PRODUCT_TYPE_CODE')]
> head(products)
  PRODUCT_NUMBER PRODUCT_TYPE_CODE
1             45                 45
2             65                 65
3             86                 86
4             101                101
5             801                 801
6             802                 802
>
> orderDetails2 <- idaMerge(orderDetails, products, 'PRODUCT_NUM
BER')
> head(orderDetails2)
Error in head(orderDetails2) :
  error in evaluating the argument 'x' in selecting a method f
or function 'head': Error: object 'orderDetails2' not found
> head(orderDetails2)
  PRODUCT_NUMBER ORDER_NUMBER PRODUCT_TYPE_CODE
1          1603           1383           1603
2          44101           1383           44101
3          44101           1383           44101
4           2232           1384           2232
5          44101           1384           44101
6          43007           1384           43007
>
> result <- idaArule(orderDetails2, tid="ORDER_NUMBER", item
="PRODUCT_TYPE_CODE", minsupport=0.01, maxlen=2, nametable =
"DASH105336.PRODUCT_TYPE", namecol = "PRODUCT_TYPE_JA")
> result
set of 2 rules
> inspect(result)
```

The graph visualization, titled "Graph for 16 rules", shows the following structure:

- Nodes: {43006}, {41013}, {1006}, {15201}, {44102}, {44101}, {41003}, {30206}, {30207}, {30203}, {2103}, {2102}, {30209}, {30208}, {30210}, {30211}, {30209,30210}, {30209,30211}
- Edges: {43006} → {1006}, {41013} → {1006}, {15201} → {1006}, {44102} → {1006}, {44101} → {1006}, {41003} → {1006}, {30206} → {30207}, {30207} → {30203}, {2103} → {2102}, {30209} → {30208}, {30209} → {30210}, {30209} → {30211}, {30208} → {30210}, {30210} → {30211}, {30209,30210} → {30211}, {30209,30211} → {30211}

Graph parameters: width: support (0.011 - 0.052), color: lift (1.699 - 13.193)

ビールを注文するモデル作成



IBM SPSS Modeler
SPSS で培った
知識が
そのまま使える

ビールを注文するモデル作成

Hbeerを
注文した人は

HBeer

ノード 0		
カテゴリ	%	n
0.000	84.942	1681
1.000	15.058	298
合計	100.000	1979

アボガドを
注文する

この
時間帯が
多い

1.000

ノード 1		
カテゴリ	%	n
0.000	40.476	17
1.000	59.524	25
合計	2.122	42

フライド
ポテトを
注文する

0.000

ノード 4		
カテゴリ	%	n
0.000	85.906	1664
1.000	14.094	273
合計	97.878	1937

hour

11.000; 12.000; 13.000; 14.000; 15.000; 16.000; 17.000 18.000; 19.000; 20.000; 21.000; 22.000; 23.000

ノード 2

カテゴリ	%	n
0.000	85.714	6
1.000	14.286	1
合計	0.354	7

ノード 3

カテゴリ	%	n
0.000	31.429	11
1.000	68.571	24
合計	1.769	35

ノード 5

カテゴリ	%	n
0.000	53.659	22
1.000	46.341	19
合計	2.072	41

ノード 8

カテゴリ	%	n
0.000	86.603	1642
1.000	13.397	254
合計	95.806	1896

FrenchFries

AhijoBucket

Predictice Analytics
ストリームを
ドラッグ & ドロップ
するだけ

The screenshot displays the Predictive Analytics interface. On the left, under 'Deployed Model Usage', a donut chart shows 'Model Predictions Remaining: 4995 of 5000' for two models: 'intro1' and 'heartland1'. Below this is a 'Manage Models - Status: Active' table:

Context Id	File	Date Created	Date Updated	Action
intro1	modelingintro.str	6/10/16	6/10/16	[Settings]
heartland1	heartland.str	6/10/16	6/10/16	[Settings]

On the right, the 'New Model Stream' section shows a 'Drop SPSS model file to' area with a 'Select File to upload' button. A large blue arrow points from this area to a workflow diagram. The diagram illustrates a data pipeline: 'heartland_list.csv' feeds into a 'Type' node, which then feeds into a 'heartLand' node. This 'heartLand' node is connected to another 'heartLand' node (with a 'c.s.0' label), which in turn feeds into 'Table' and 'Analysis' nodes.

IBM Ready? Try the new Bluemix | New! Try OpenWhisk

ダッシュボード ソリューション カタログ

'Second Trial' のダッ... ←

.js 環境変数

HandsOn001

- 概要
- SDK for Node.js™
- ファイル
- ログ
- 環境変数 >**
- コーディングの開始

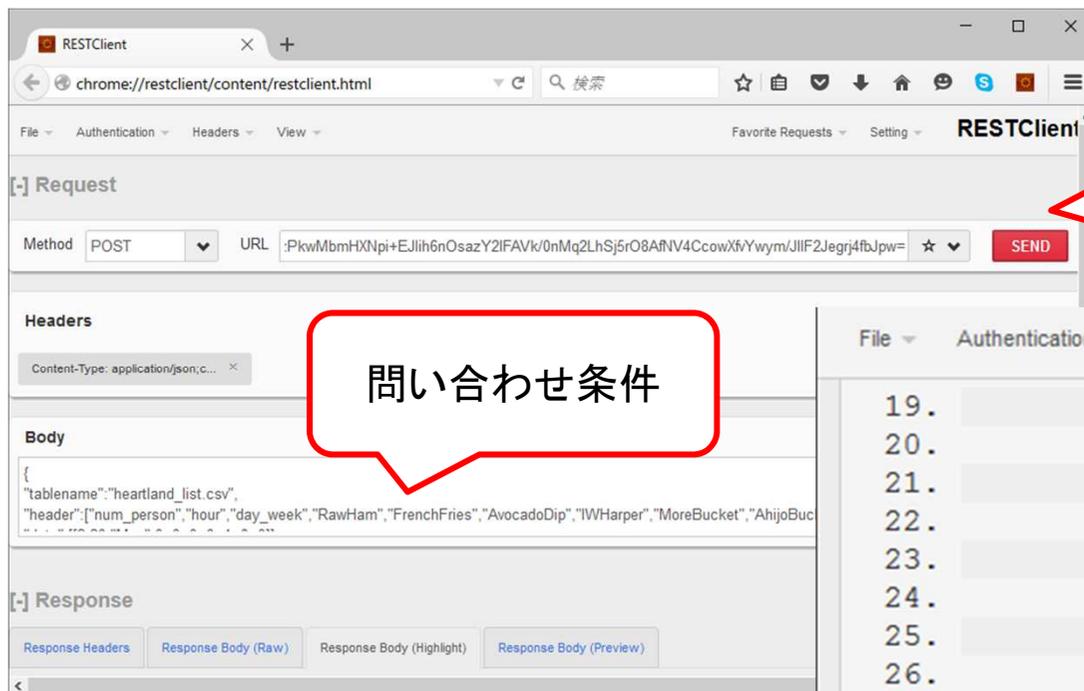
サービス

- dashDB
- Predictive Analytics

```
{
  "pm-20": [
    {
      "name": "Predictive Analytics-gp",
      "label": "pm-20",
      "plan": "Free",
      "credentials": {
        "access_key": "iFX4ABnvh2GJMDpLz1x+SKqFotreCVy0/aLZ0rH0H",
        "url": "https://palbyp.pmservice.ibmcloud.com/pm/v1"
      }
    }
  ],
  "dashDB": [
    {
      "name": "dashDB-xr",
      "label": "dashDB",
      "plan": "Entry",
      "credentials": {
        "port": 50000,
        "db": "BLUDB",

```

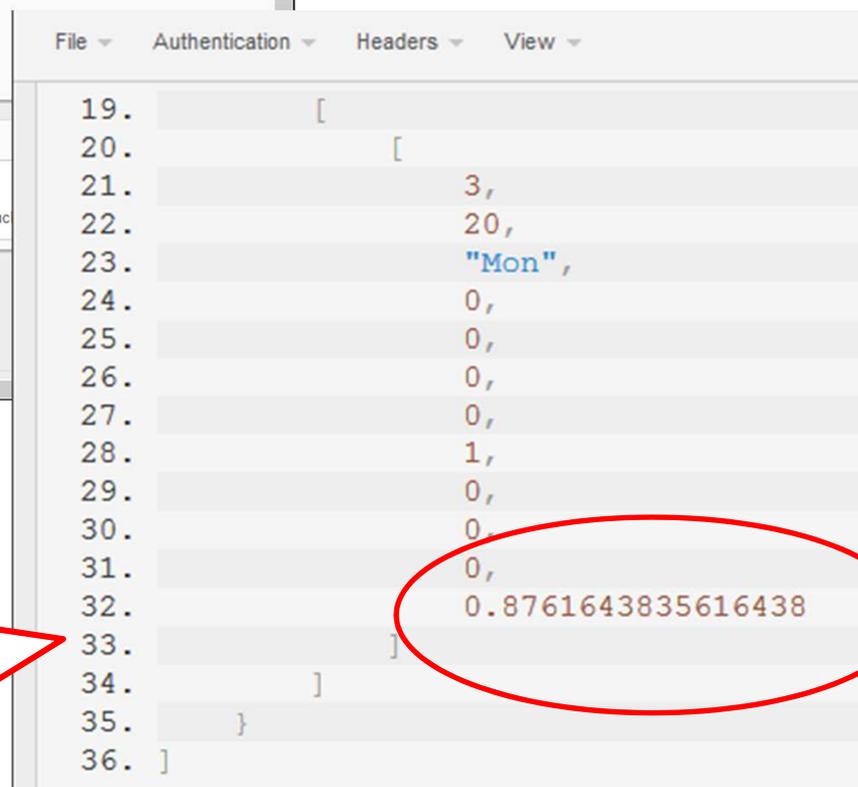
接続情報(JSON)
が発行される



問い合わせ条件

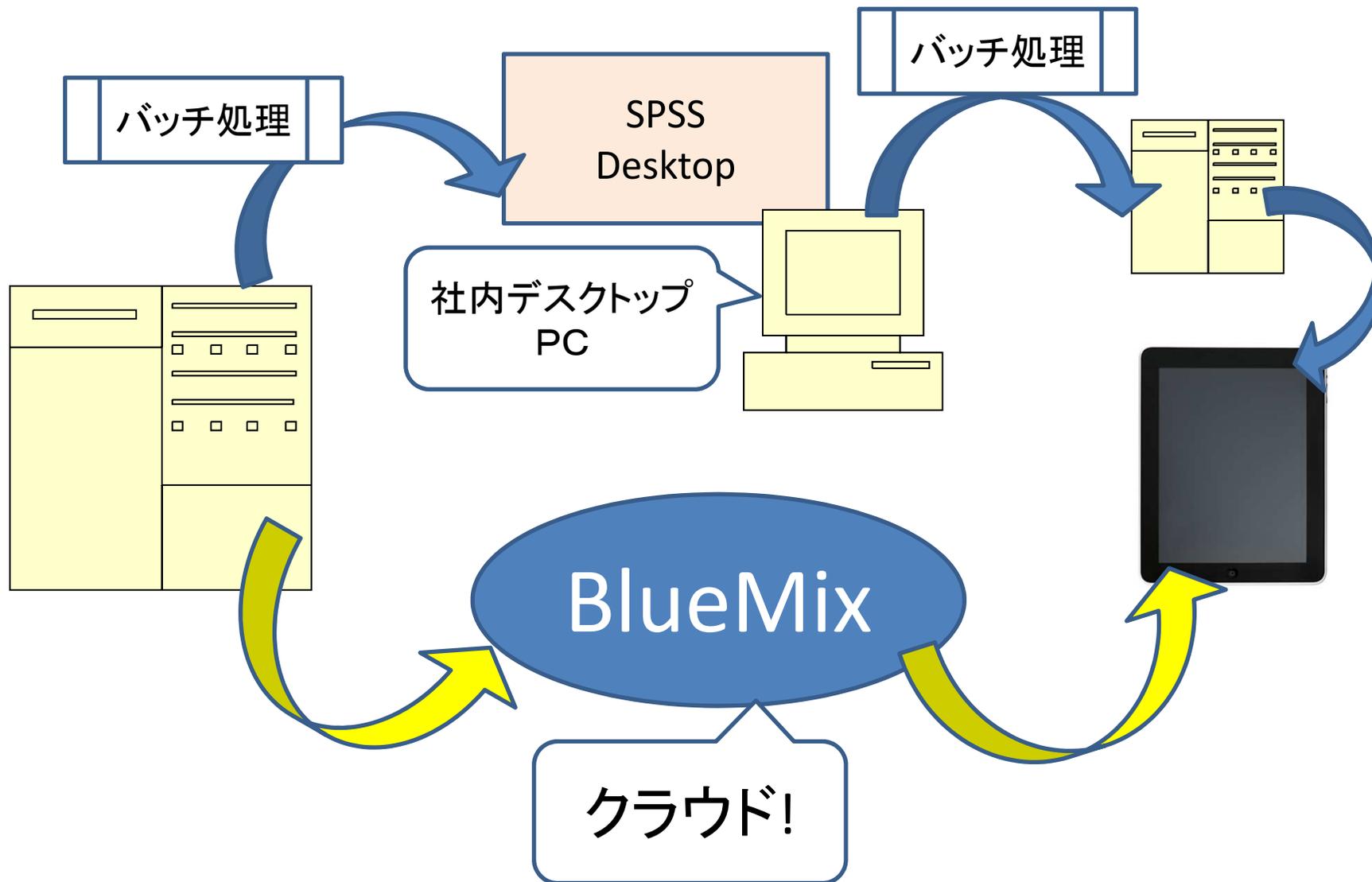
ブラウザから
アクセスすると...

...注文する
確率が得られる



アクティブトラック解析

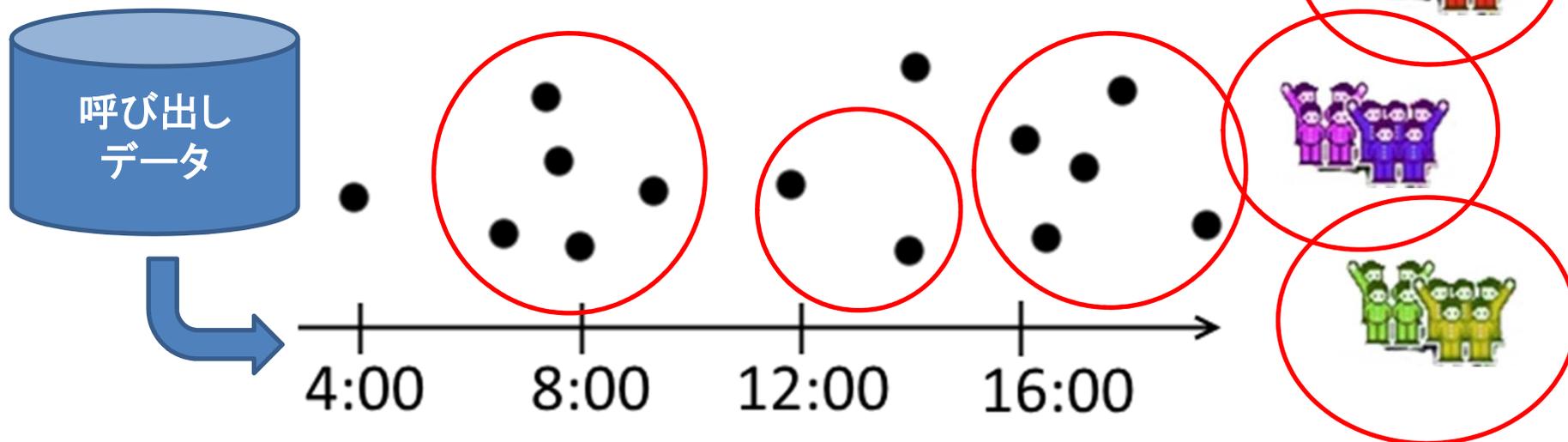
環境省 平成23年度地球温暖化対策技術開発等事業
平成24年度地球温暖化対策技術開発・実証研究事業
「EVタクシーの実用化促進と
運用方法確立のための実証研究」



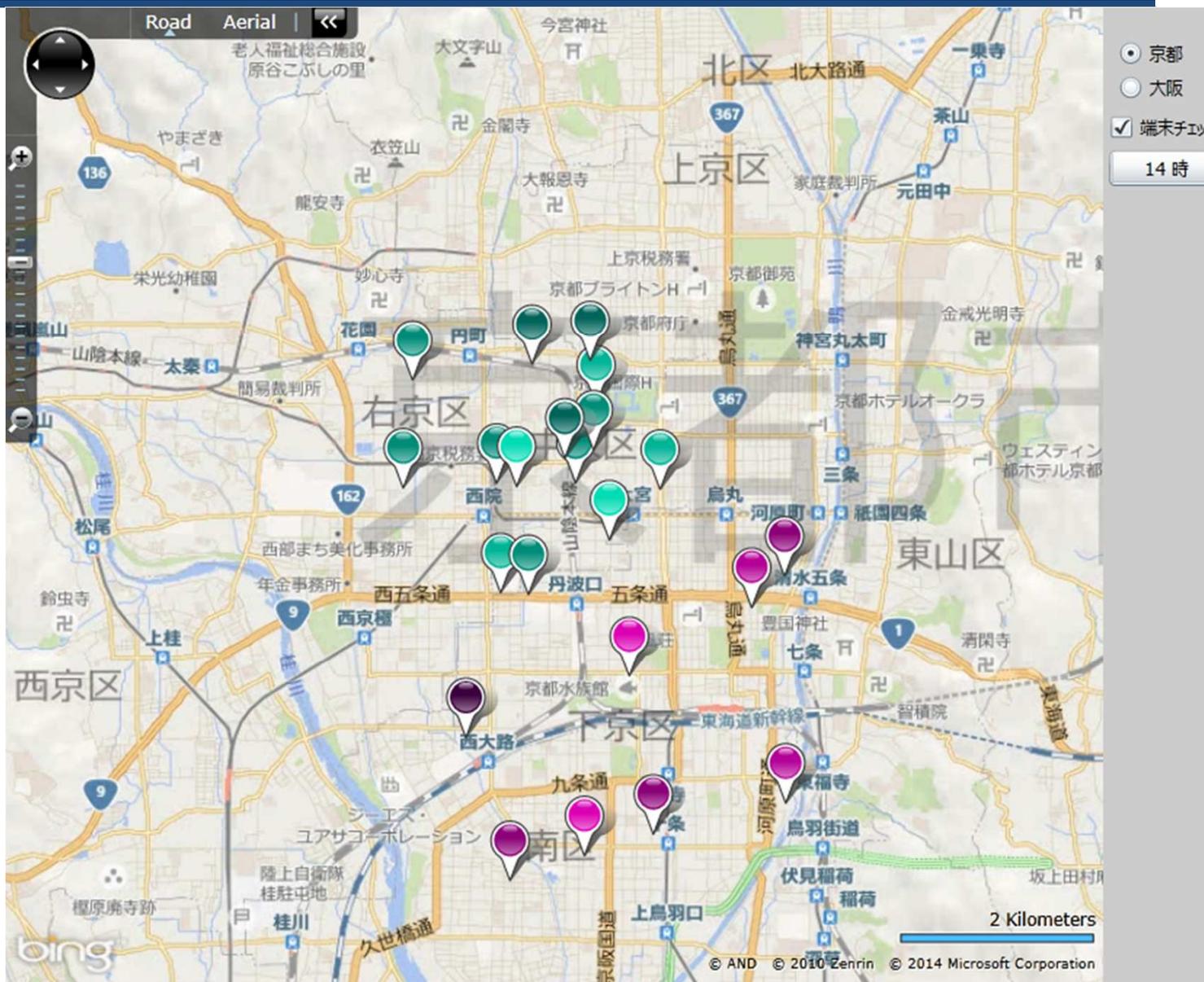
タクシー呼び出しスポットの抽出

- ・呼び出しデータを(空間×時間)でクラスタリング
- ・クラスタの大きさを5段階評価
- ・「呼び出し地点をぼかして欲しい」との要望
- ・クラスタ数は手動で調整

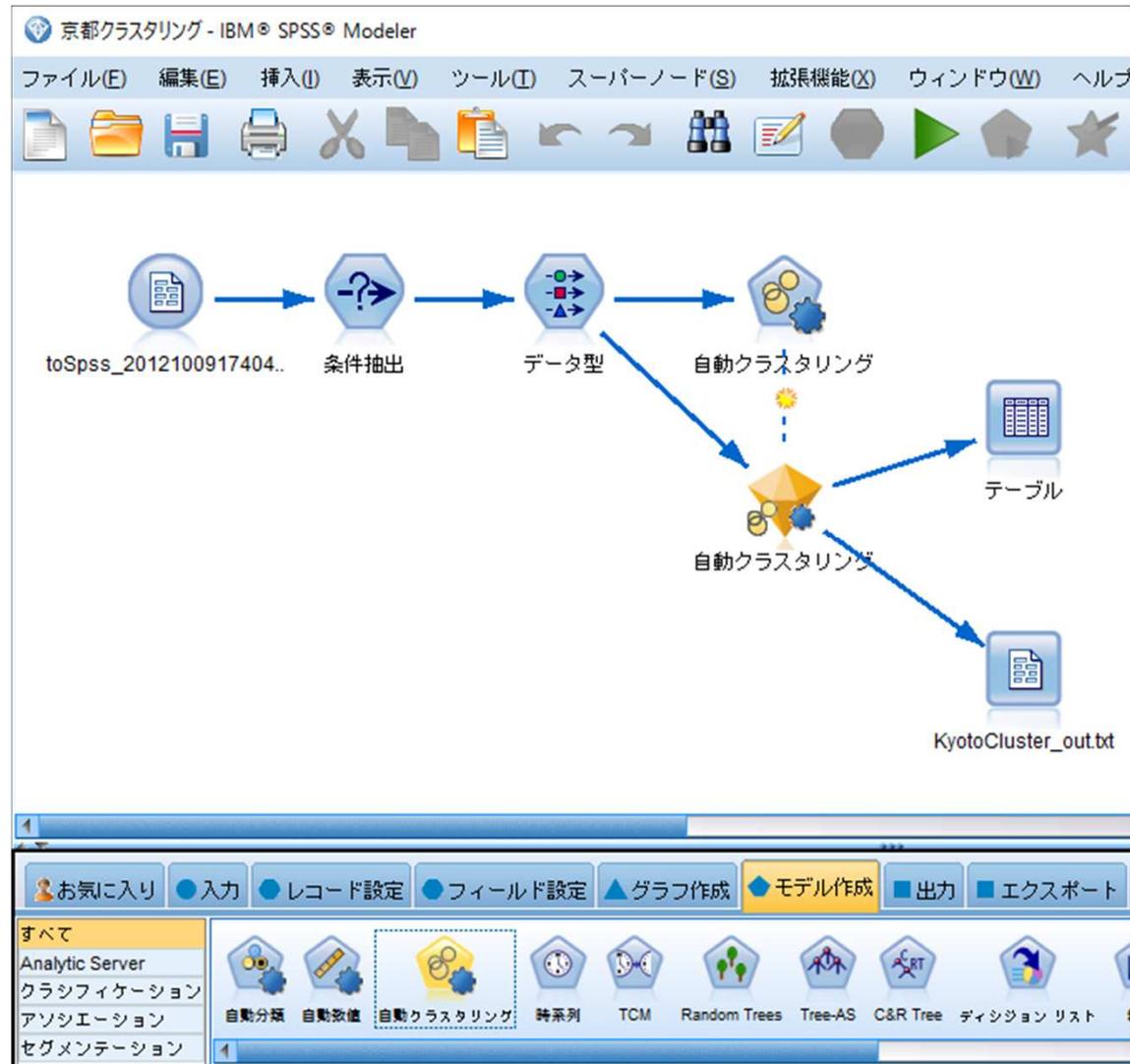
-- 京都エリア=300, 大阪エリア=750



タクシー呼び出し状況



京都のタクシー呼び出し地域をまとめる



京都のタクシー呼び出し地域をまとめる

ツールが
あらゆる方法を
試す

自動クラスタリング

ファイル(E) 生成(G) プレビュー(P)

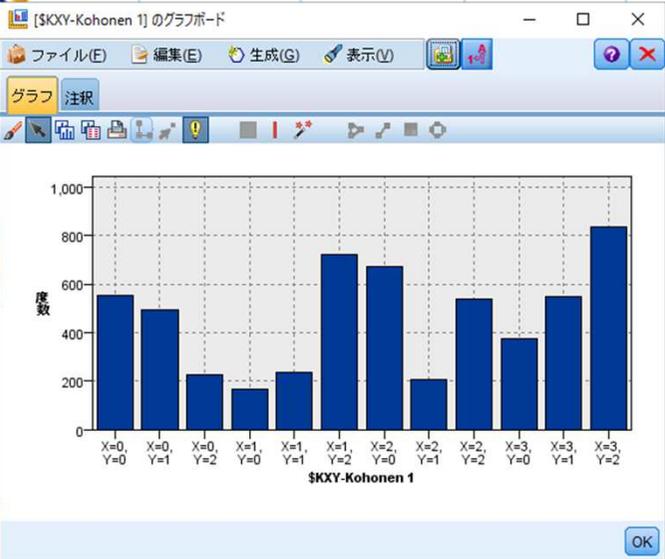
モデル 要約 注釈

ソート項目(S): 使用 昇順 降順 未使用モデルの削除 表示: 学習セット

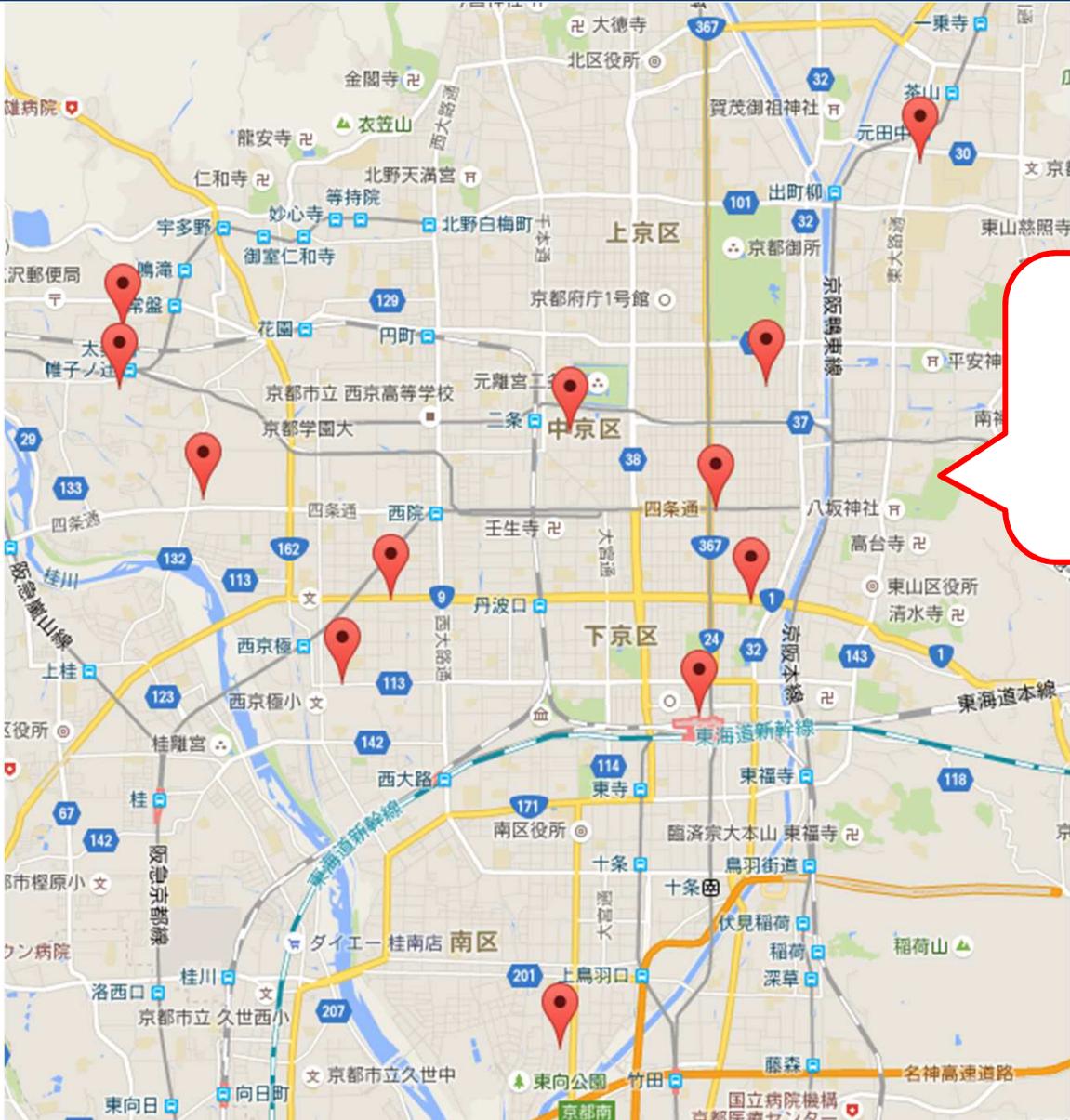
使...	グラフ	モデル	構築時間 (分)	全体像	クラスター 数	最小 クラスター	最小 クラスター	最大 クラスター	最大 クラスター	最小/ 最大	重要度
<input type="checkbox"/>		Two... < 1		0.644	3	6	0	3988	71	0.002	0.0
<input type="checkbox"/>		K-m... < 1		0.585	5	2	0	2618	46	0.001	0.0
<input checked="" type="checkbox"/>		Koh... < 1		0.432	12	166	2	834	14	0.199	0.0

OK キャンセル

適用(A) リセット(R)

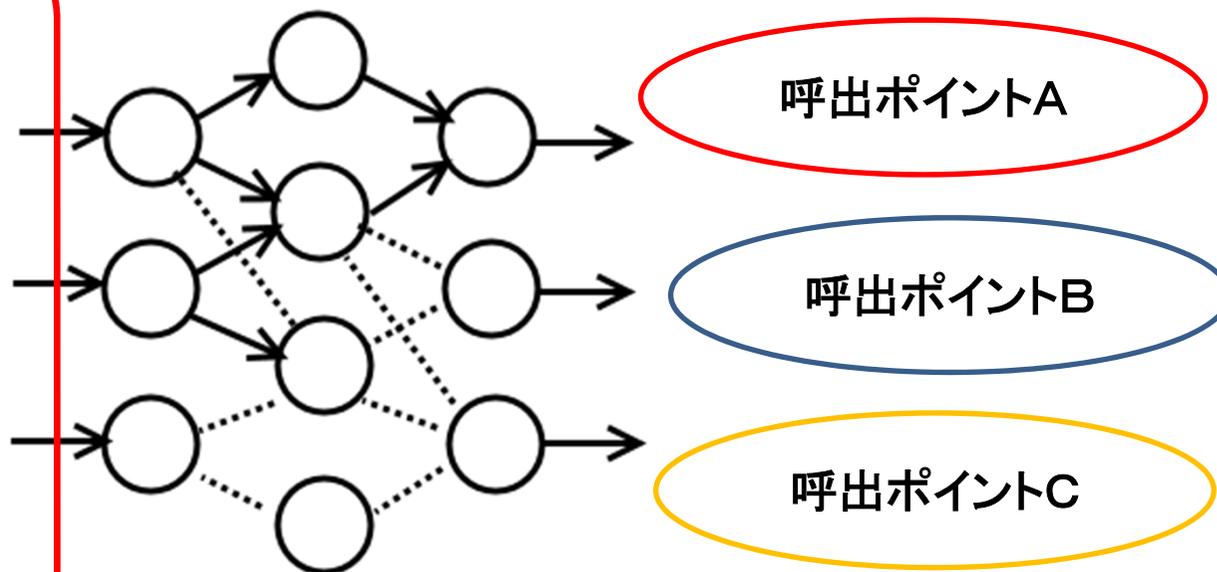


京都のタクシー呼び出し地域をまとめる



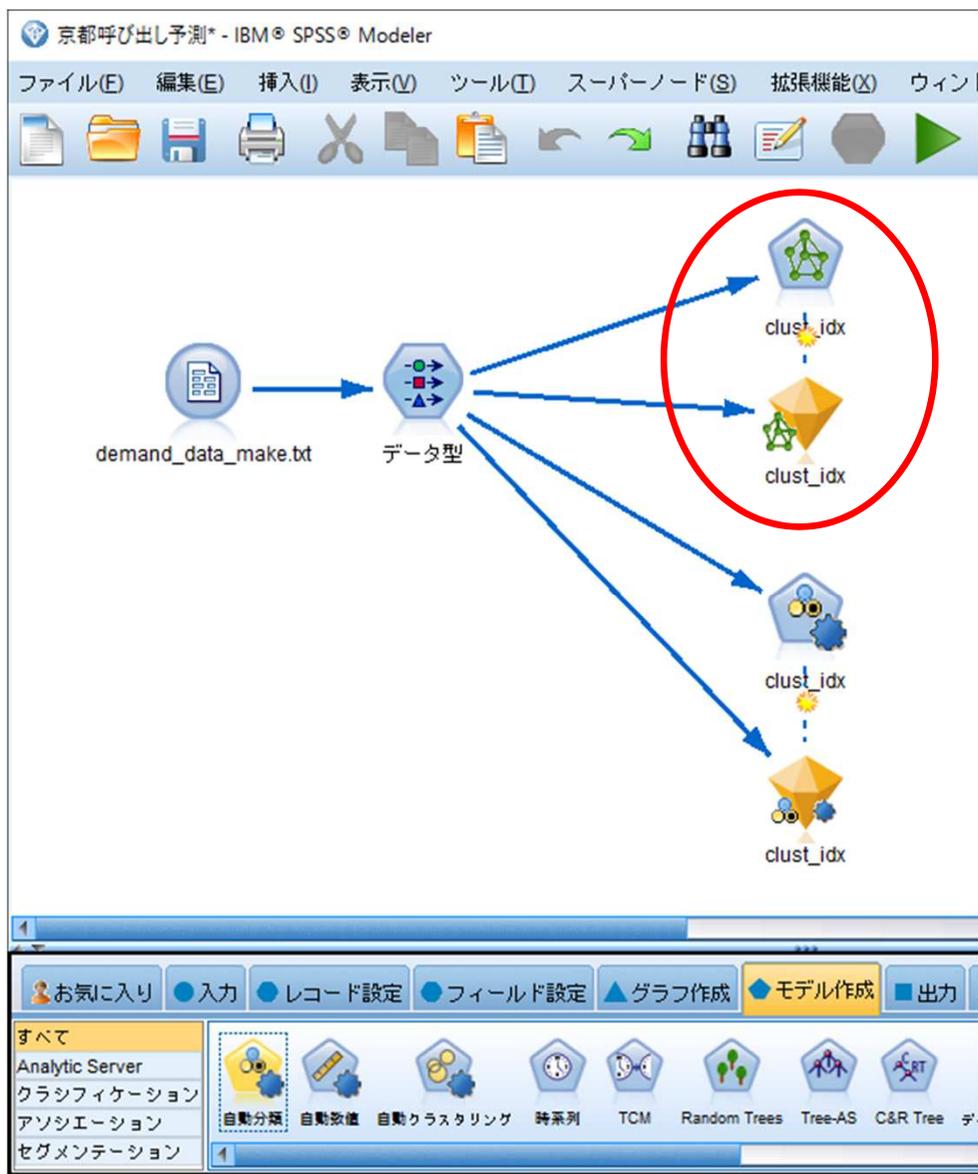
今回は
12スポットを
抽出

- ・時刻
- ・曜日、平日/休日
- ・季節
- ・天候、気温
- ・呼び出し状況
- ・端末チェック状況
- ・(イベントの有無)



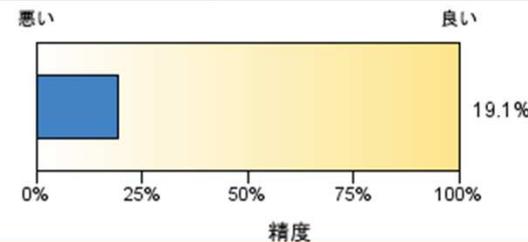
[SPSSニューラルネットワークモジュール]

京都のタクシー呼び出し需要予測



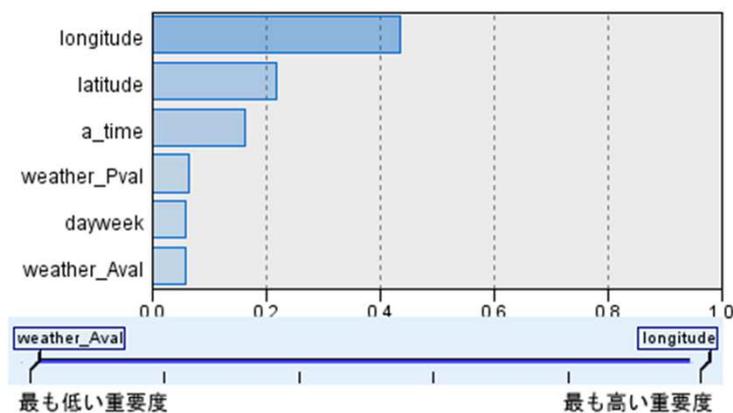
モデル要約

目標	clust_idx
モデル	多層パーセプトロン
使用される停止規則	誤差を小さくすることができません
隠れ層 1 ニューロン	13

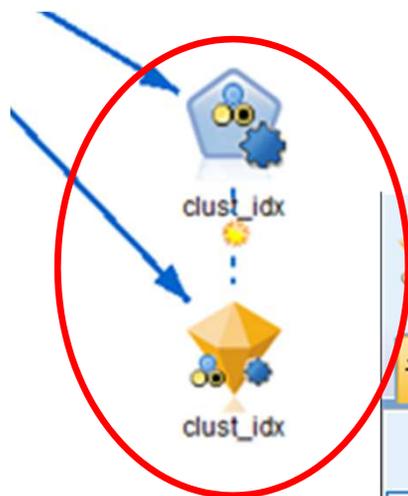


予測変数の重要度

対象: clust_idx



もっと良い
方法は
無いものか



ファイル(E) 生成(G) 表示(V) プレビュー(P)

モデル グラフ 要約 設定 注釈

ソート項目(S): 使用 昇順 降順 未使用モデルの削除 表示:

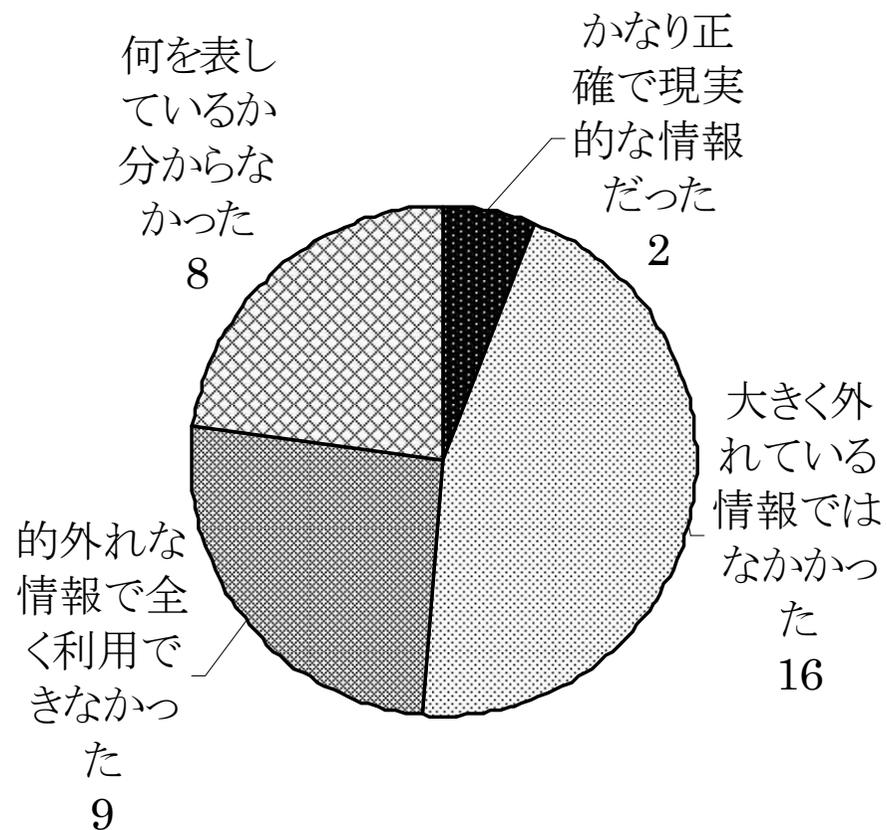
使用?	グラフ	モデル	構築時間(分)	全体精度(%)	使用フィールド数
<input checked="" type="checkbox"/>		C5.1	< 1	97.627	3
<input type="checkbox"/>		C&R Tree 1	< 1	86.229	5
<input type="checkbox"/>		CHAID 1	< 1	84.120	2

この辺で
呼んでるよ！

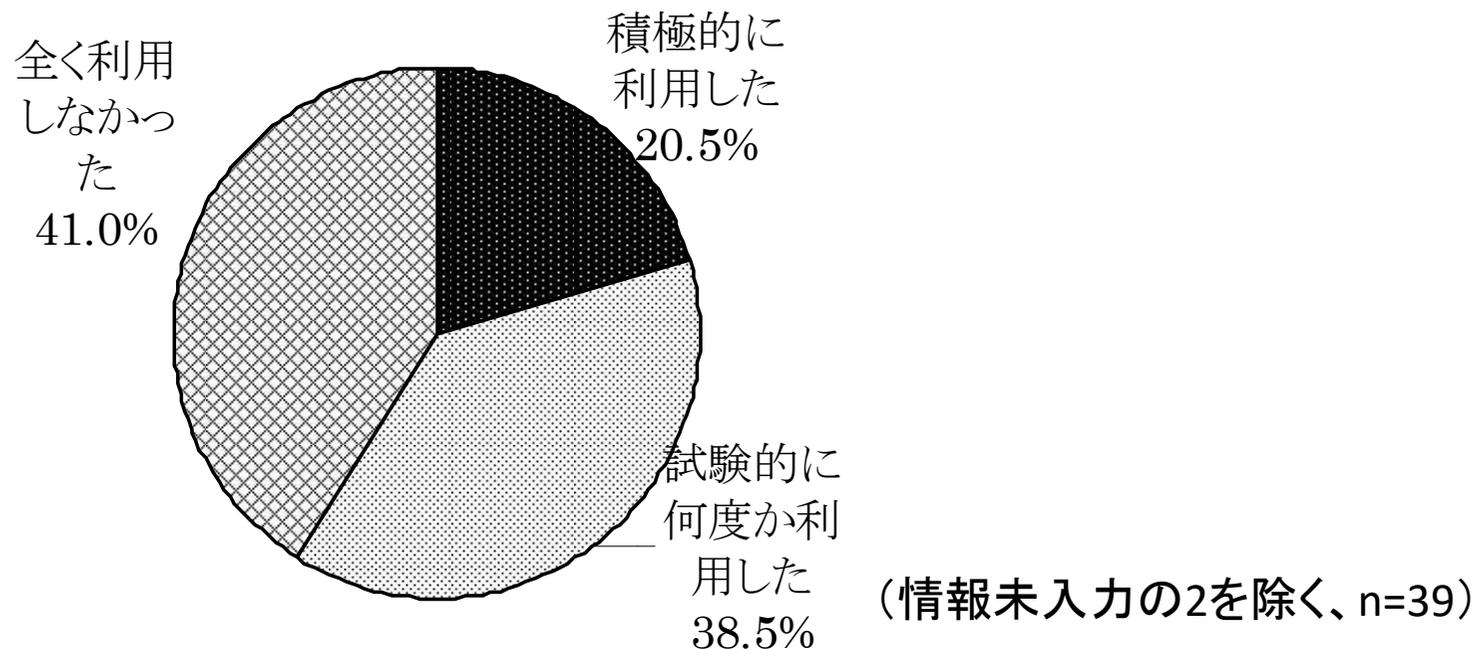


“手を挙げる”アイコンで
呼び出し需要を5段階表示

■ 需要予測機能に対する評価



■ 呼び出し・応答以外の機能の利用状況



- ・予測外れと評価した回答者の中で6名(全9名の3分の2)は、呼び出し・応答以外の機能に関する利用状況について、「全く利用しなかった」と回答。
- ・利用していたと回答した者における予測外れの評価は、回答者の約14%。

EVタクシー車両の走行状況(一日あたり)



	走行距離 (km)	うち実車 距離 (km)	運送回数 (回)	輸送人員 (人)	営業収入 (税込) (円)	実車率
導入前 (2011/12/1~2012/1/29)	99.4	37.0	9.8	13.9	13,440	37.0%
導入後 (2012/12/1~2013/1/29)	95.5	36.3	10.3	14.6	13,674	38.2%
導入効果(／日・台)	-3.8	-0.7	+0.5	+0.7	+234	+1.2%
導入後 (2012/10/1~2013/2/28)	99.7	39.0	10.6	15.0	14,454	39.1%
導入効果(／日・台)	0.3	2.0	+0.8	+1.2	+1,014	+2.1%

[IoT]

加速度センサの活用

“幸せは、 加速度センサで 測れる”

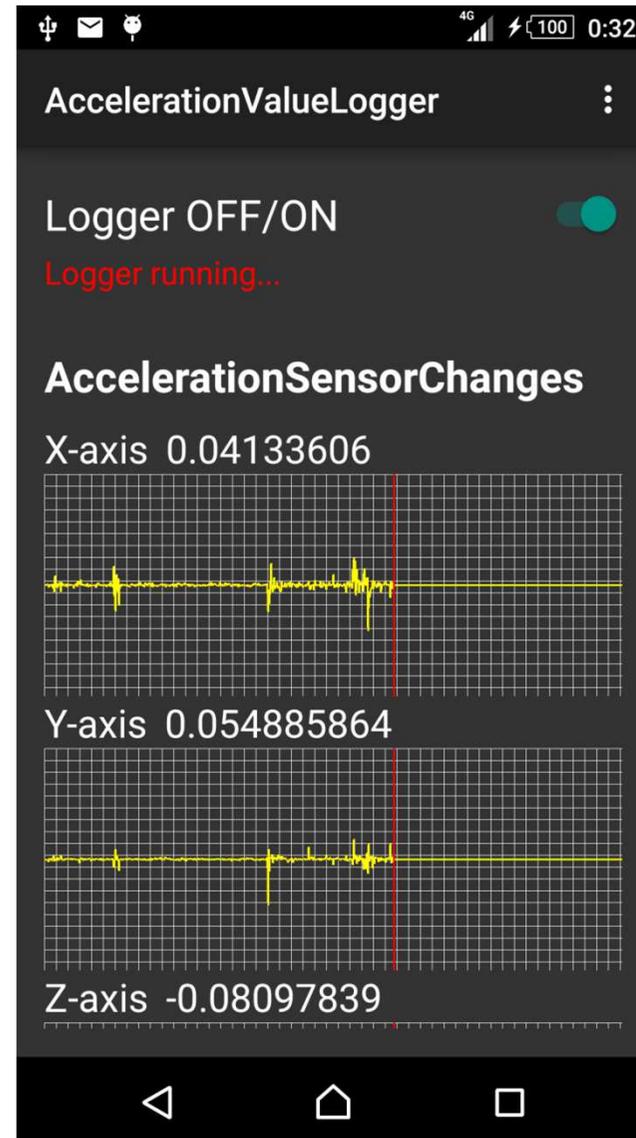
-- 日立中央研究所 矢野和男



加速度センサーで測ってみた



1. 休日の自宅
2. 休日のサイクリング
3. 休日のジョギング
4. 『授業中』
5. 会社でデスクワーク



“R-Studio”上で「活発度」を解析

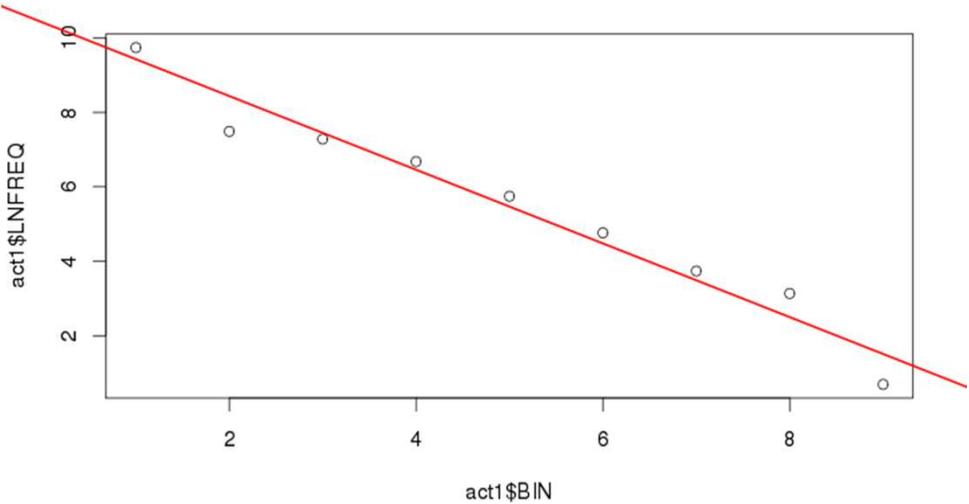
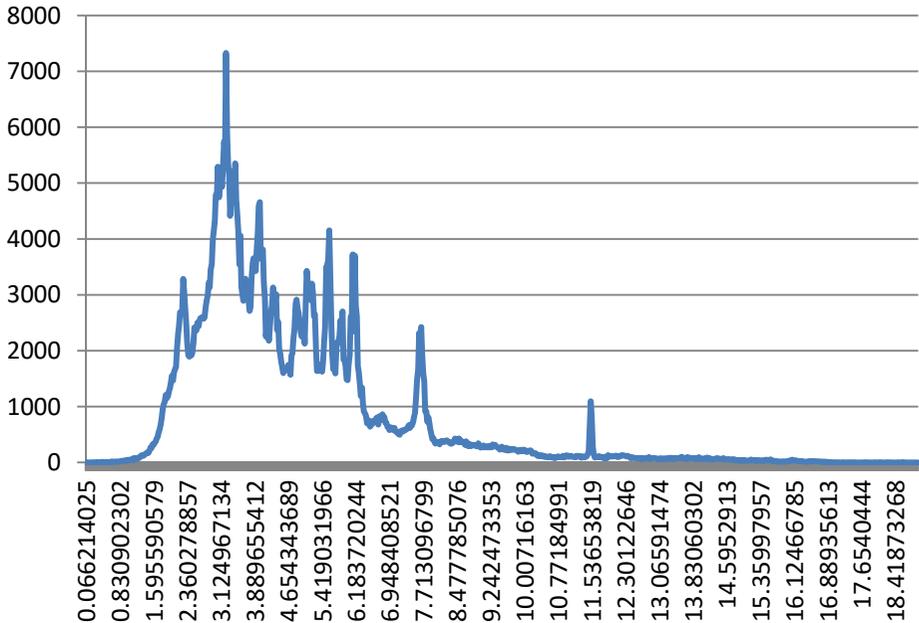
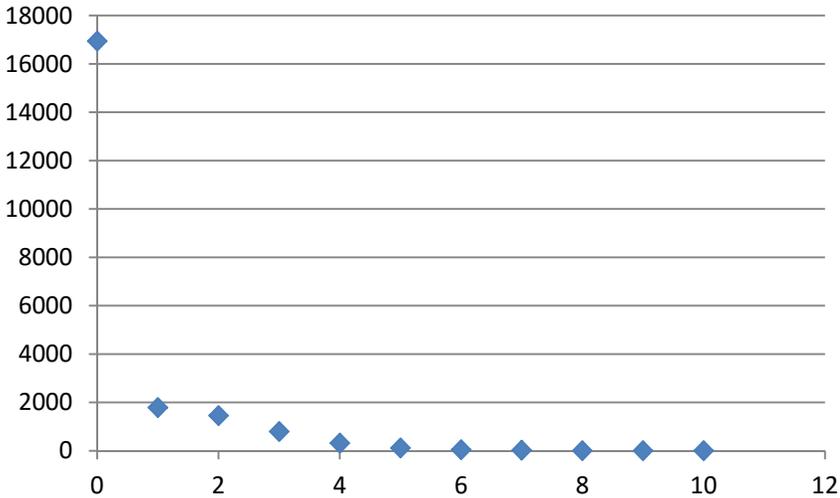
The screenshot shows the R-Studio interface. The console on the left contains the following R code and output:

```
(Intercept)      LNFREQ  
9.8181         -0.9011  
  
> plot(act2$BIN, act2$LNFREQ )  
> abline(act2.lm, lwd=2, col="red")  
>  
> act3.lm <- lm( BIN~LNFREQ, data=act3 )  
> act3.lm  
  
Call:  
lm(formula = BIN ~ LNFREQ, data = act3)  
  
Coefficients:  
(Intercept)      LNFREQ  
7.2894         -0.6548  
  
> act4.lm <- lm( BIN~LNFREQ, data=act4 )  
> act4.lm  
  
Call:  
lm(formula = BIN ~ LNFREQ, data = act4)  
  
Coefficients:  
(Intercept)      LNFREQ  
9.4362         -0.9417  
  
>  
>  
> act5.lm <- lm( BIN~LNFREQ, data=act5 )  
> aact5.lm  
Error: object 'aact5.lm' not found  
> act5.lm  
  
Call:  
lm(formula = BIN ~ LNFREQ, data = act5)  
  
Coefficients:  
(Intercept)      LNFREQ  
12.510         -1.391  
  
> |
```

The plot on the right shows a scatter plot of `act1$LNFREQ` (y-axis) versus `act1$BIN` (x-axis). The y-axis ranges from 2 to 10, and the x-axis ranges from 0 to 10. There are approximately 12 data points represented by open circles. A solid red regression line is drawn through the points, showing a clear negative linear correlation. The plot is titled "act1\$LNFREQ" on the y-axis and "act1\$BIN" on the x-axis.

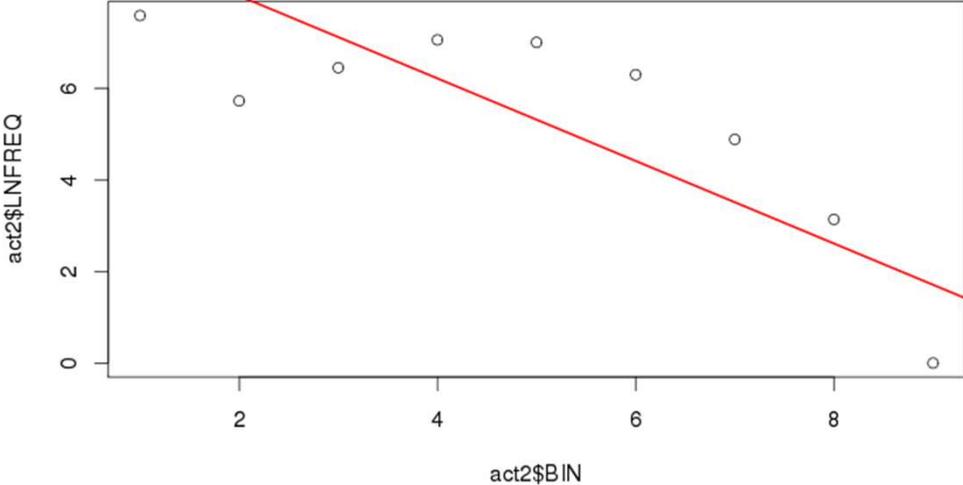
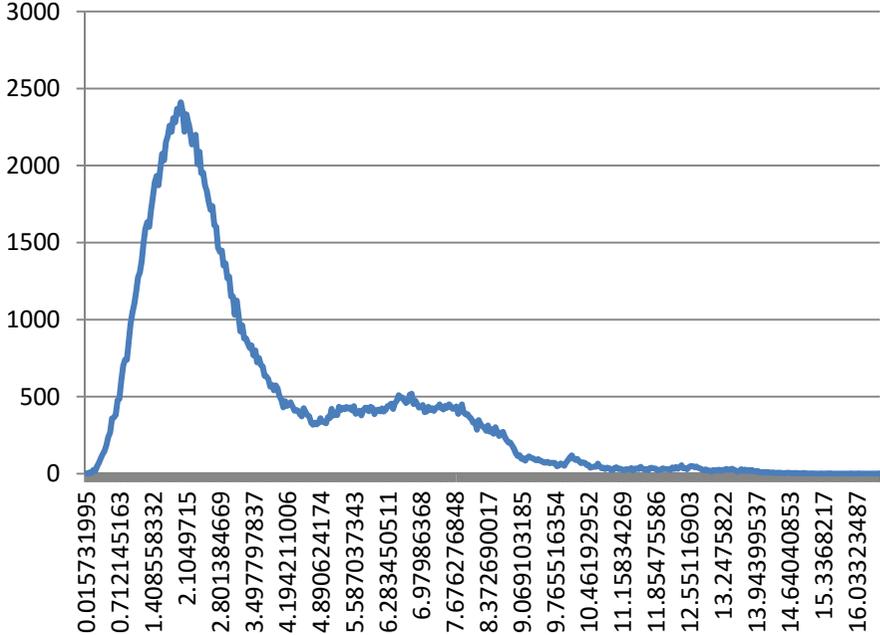
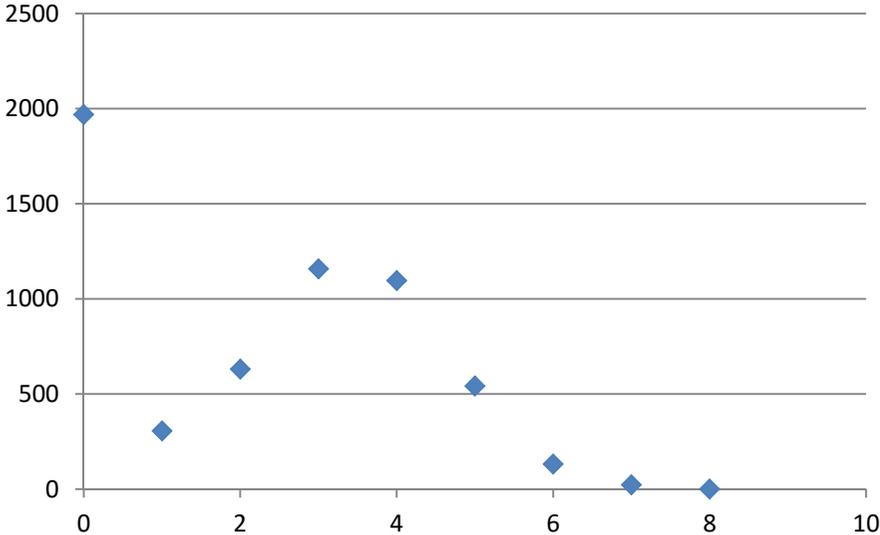
グラフが少々
貧弱だった・・・
(R言語の特性)

1. 休日の自宅



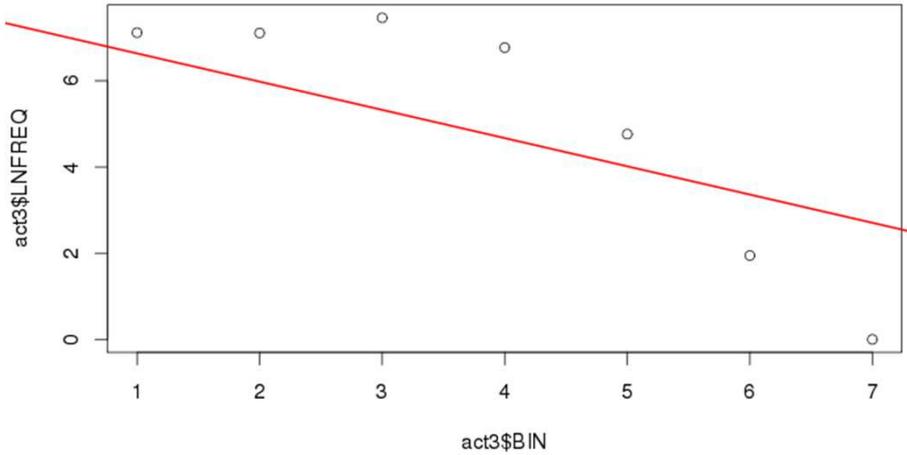
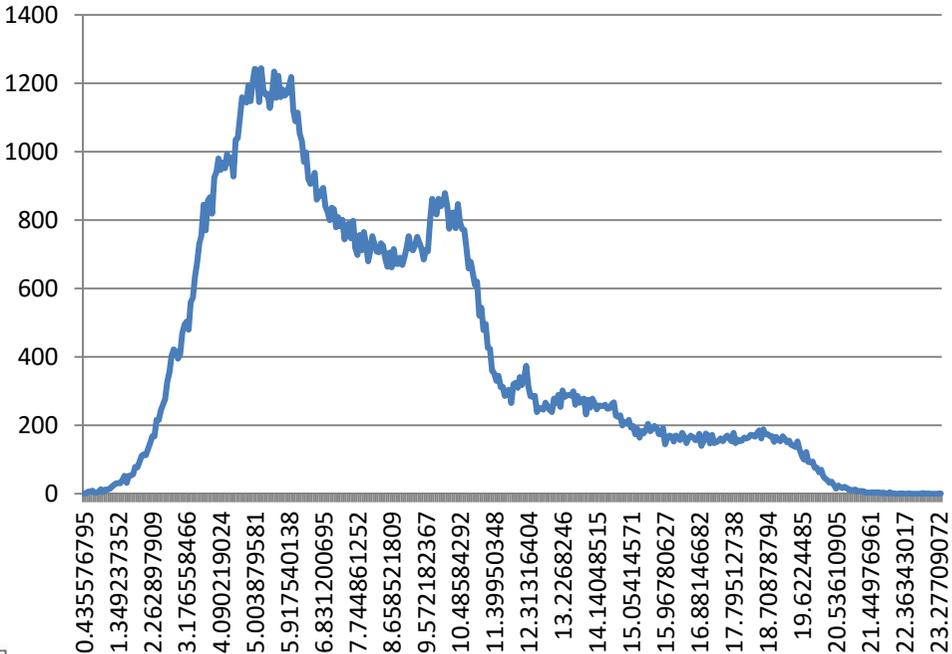
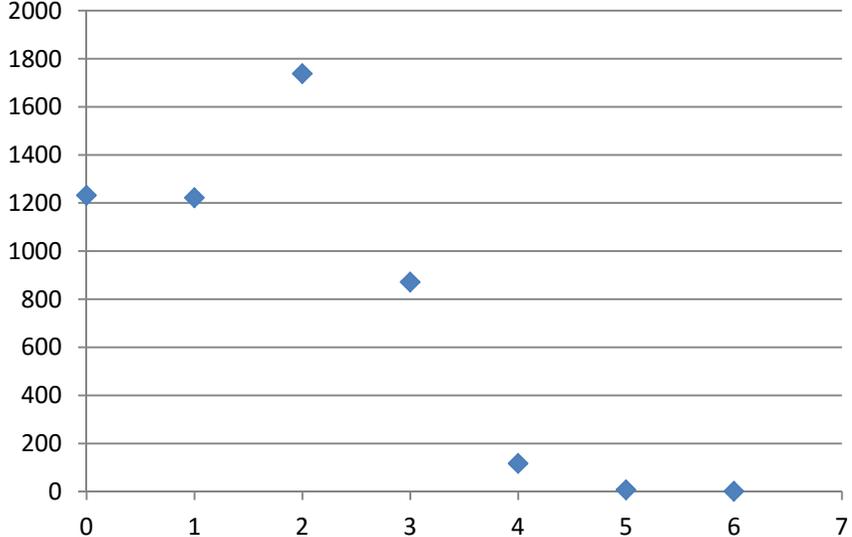
Coefficients:
 (Intercept) LNFREQ
 10.4145 -0.9893

2. 休日のサイクリング



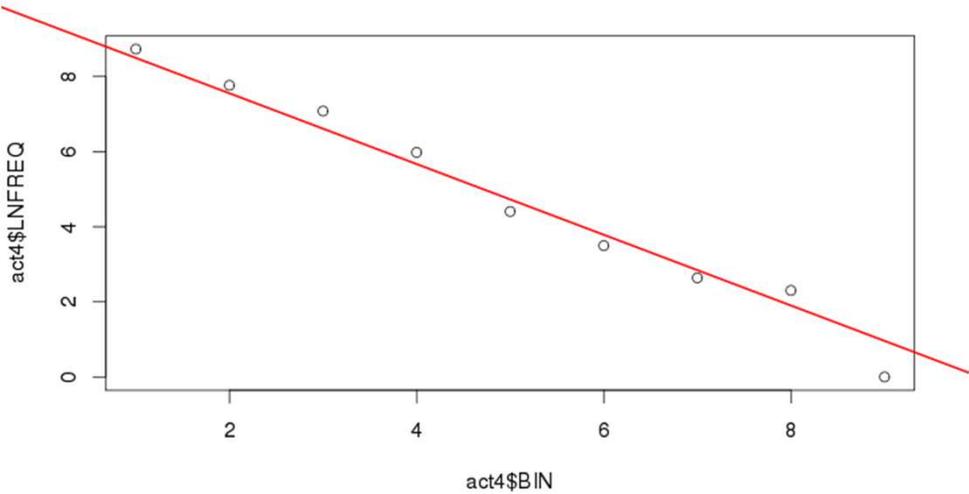
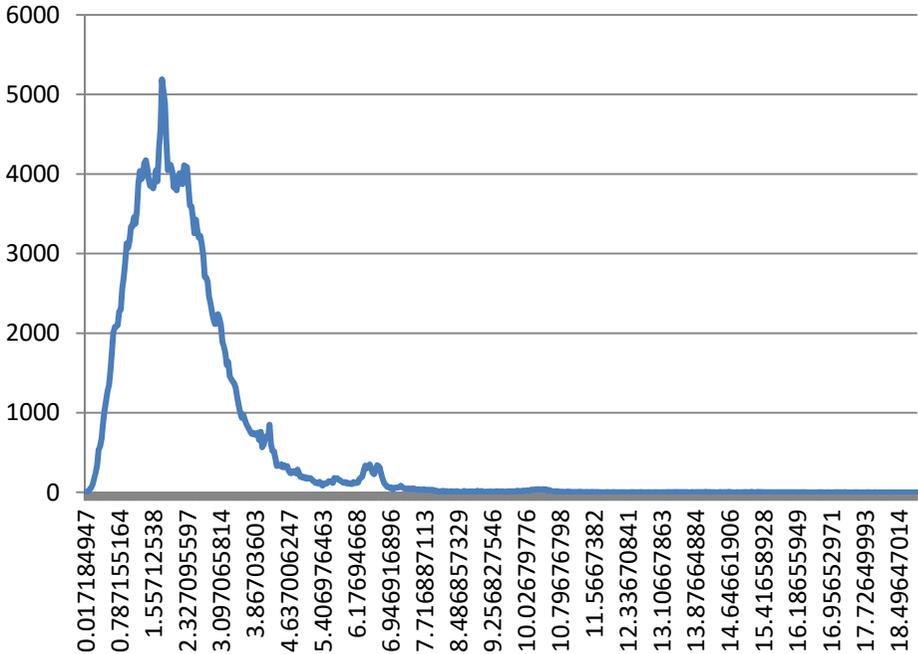
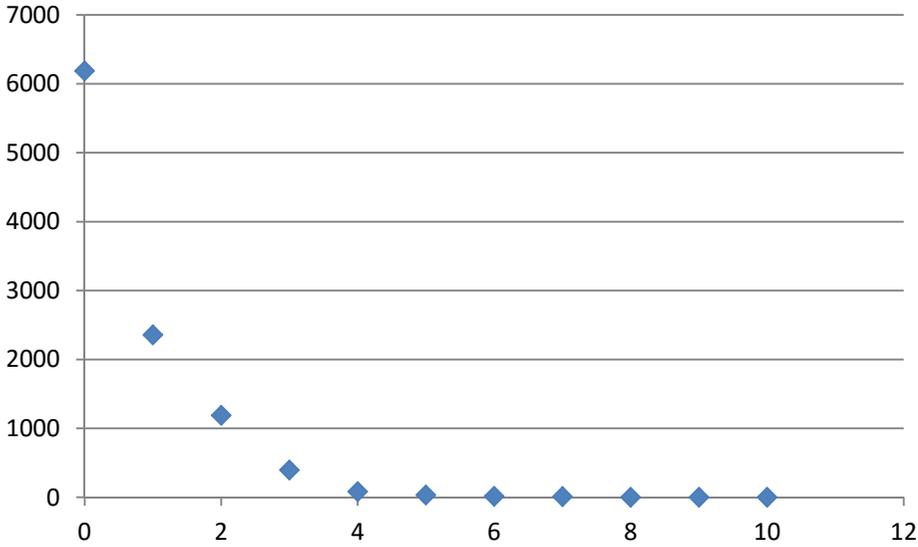
Coefficients:
 (Intercept) LNFREQ
 9.8181 -0.9011

3. 休日のジョギング



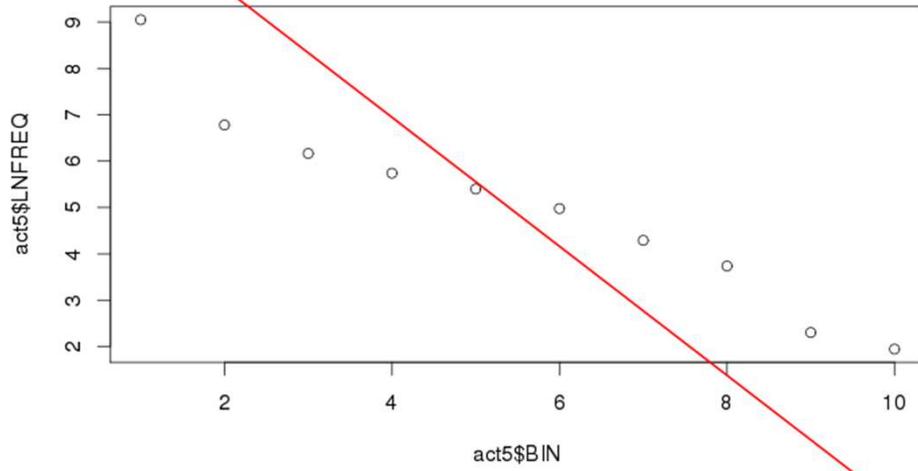
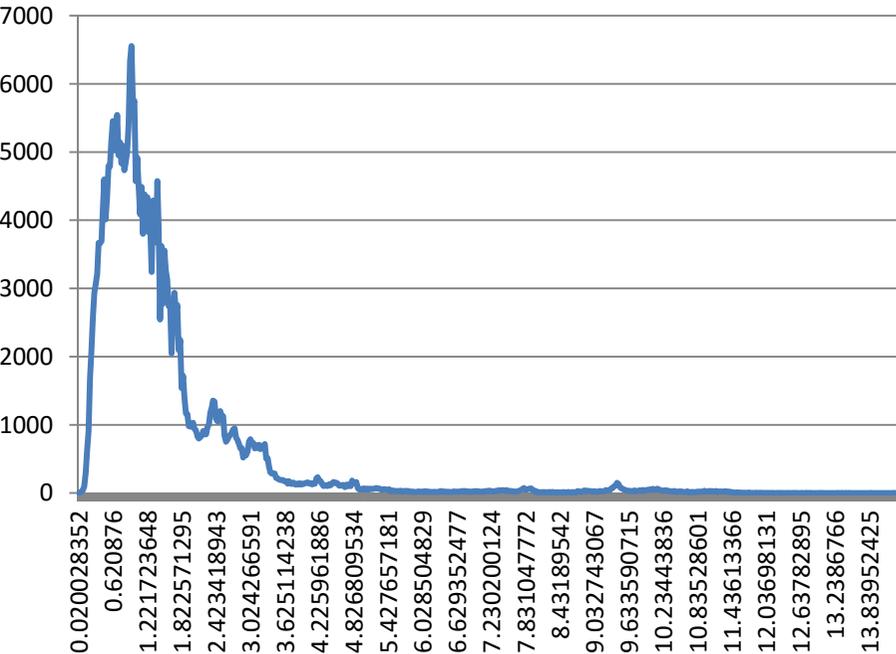
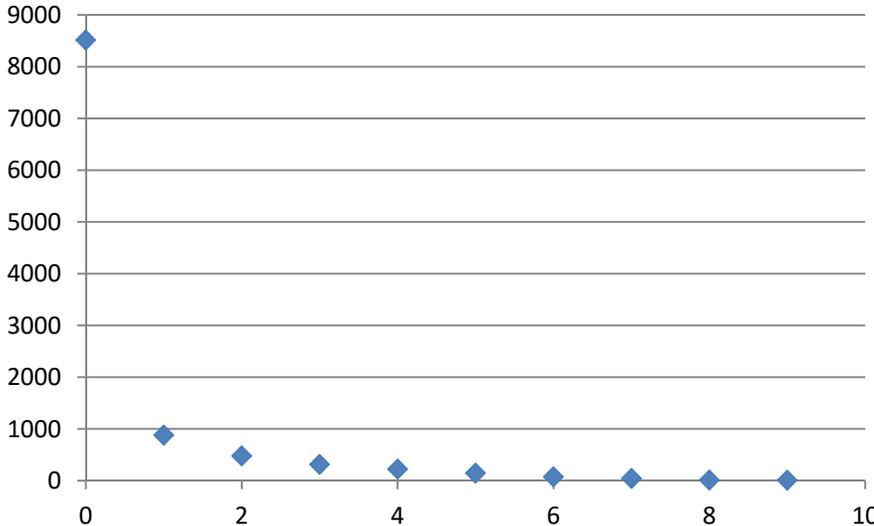
Coefficients:
 (Intercept) LNFREQ
 7.2894 -0.6548

4. 『授業中』



Coefficients:
 (Intercept) LNFRQ
 9.4362 -0.9417

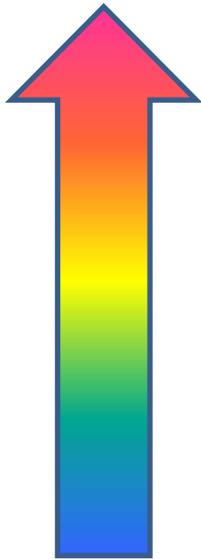
5. 会社でデスクワーク



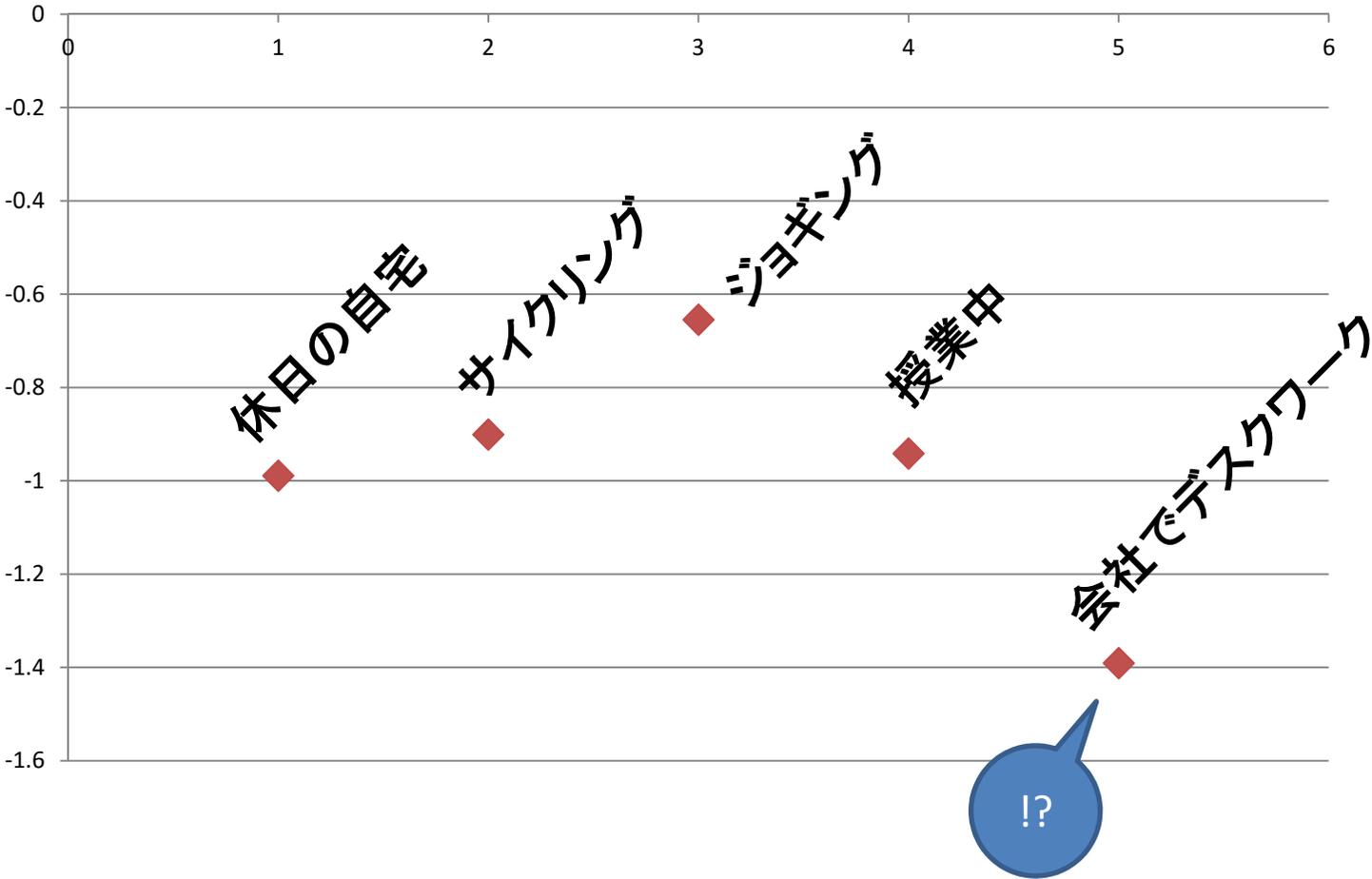
Coefficients:
 (Intercept) LNFREQ
 12.510 -1.391

1. 休日の自宅

活動度
(高)



(低)



以上、
はじめてBlueMixに触れてから
3日でできた。

- 違和感なく導入できる
[コスト面 / 技術面]
- 高い生産性
- 連携がすごい

- 本家以外の
資料が少ない