データの分析と知識発見

ルールの抽出

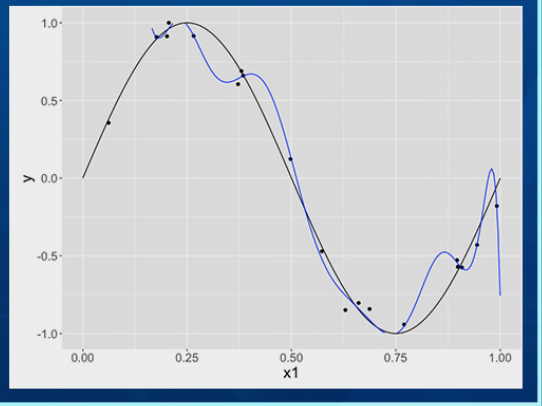
- 電話帳からいくつかの電話番号と名前を覚える

  ➡ 学習していない人の電話番号はわからない
  　電話番号にルールがない

- 気温や湿度から海の家の売上を予測する
- 過去数日の天気から次の日の天気を予測する

  ➡ 何らかのルールに従って
  　変動しているものに対して用いる

- 背後にある確率分布を想定する
- 多項式　三角関数などで近似する
- 機械学習



最尤(さいゆう)推定(1)

◆ 最も尤(もっと)もらしい推定量

確率 p で起こるベルヌイ試行を n 回行うときに起こる回数は

二項分布　$P(X=k)={}_nC_k p^k(1-p)^{n-k}$

5回の試行を行って3回という結果が観測されたとすると

$$P(X=3)={}_5C_3 p^3(1-p)^2=10(p^5-2p^4+p^3)$$

の確率の出来事が起きたと考えることができる

$p=\frac{1}{3}$ ならば　$P(X=3)=\frac{40}{243}=0.1646\cdots$

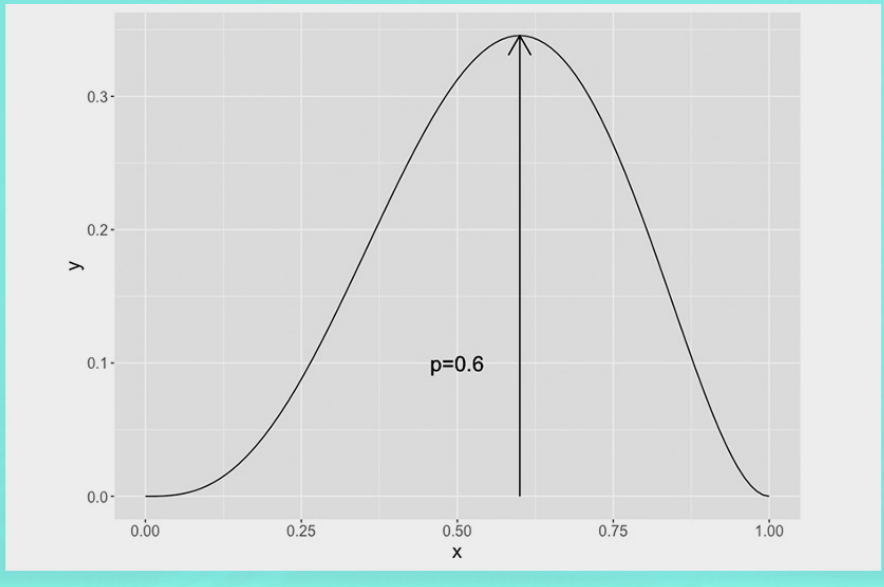
$p=\frac{1}{2}$ ならば　$P(X=3)=\frac{5}{16}=0.3125$

確率が大きい方が
起きたと考える方が
尤もらしい

最尤(さいゆう)推定(2)

◆ 5回の試行を行って3回という結果が観測されたとすると

$$P(X=3) = {}_5C_3\, p^3(1-p)^2 = 10(p^5 - 2p^4 + p^3)$$



この値が最大になる

$$p = 0.6$$

が最も尤(もっと)もらしい推定量

（最尤推定量）

最尤推定(3)

単回帰分析 $Y = aX + b + \varepsilon$ において ε が正規分布 $N(0, \sigma^2)$ に従うとする

n 組のデータ $(x^{(i)}, y^{(i)})$ $(i = 1,2,\cdots n)$ があるとすると

予測誤差 $y^{(i)} - (ax^{(i)} + b)$ の確率密度関数は

$$p^{(i)} = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y^{(i)} - ax^{(i)} - b)^2}{2\sigma^2}\right) \quad \text{から}$$

尤度関数 $L(a,b) = \prod_{i=1}^{n} p^{(i)} = p^{(1)} \cdot p^{(2)} \cdots p^{(n)}$ を考える

$-\log L(a,b)$ を計算すると

$$-\log L(a,b) = n \log \sqrt{2\pi}\sigma + \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y^{(i)} - ax^{(i)} - b)^2$$

$$\log(xy) = \log x + \log y$$

Q-Q plot

正規分布かどうかを確認するには

❶ ヒストグラムを作成して目で確認
❷ Q-Q plot を作成する

n 個のデータを小さい順に並べた場合 一様分布であれば

i 番目までの割合は $\frac{i}{n}$

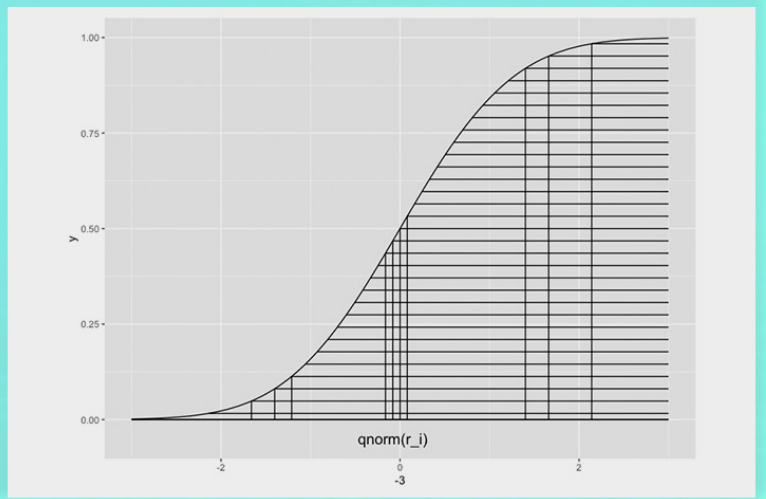
このまま計算すると全体が1となるので

R では

$$r_i = \frac{i - a}{n + 1 - 2a}$$

と補正して累積分布の x 座標を求めて比較

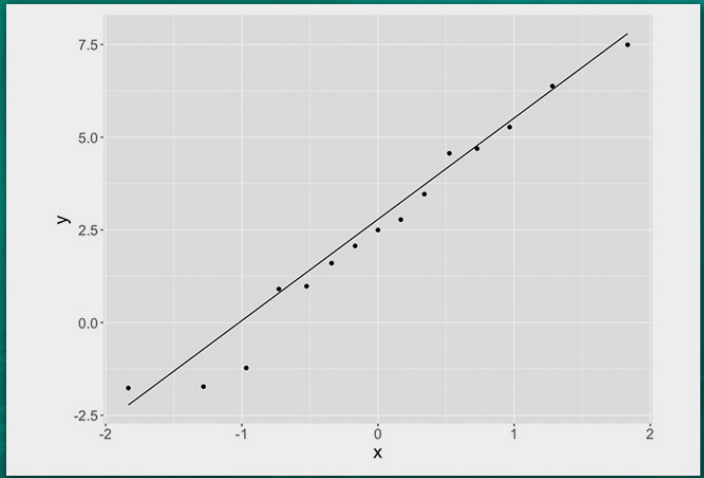
(n >10 のとき a =0.5)



RでのQ-Q plot

```
> set.seed(15)
> x <- rnorm(15,2,3)
> ggplot(mapping=aes(sample=x))
+     geom_qq()+ geom_qq_line()
```

```
# sample でデータを指定
# 平均と分散が違う場合 直線の傾き等が変わる
# データが直線上に並ぶかどうかで判定
```



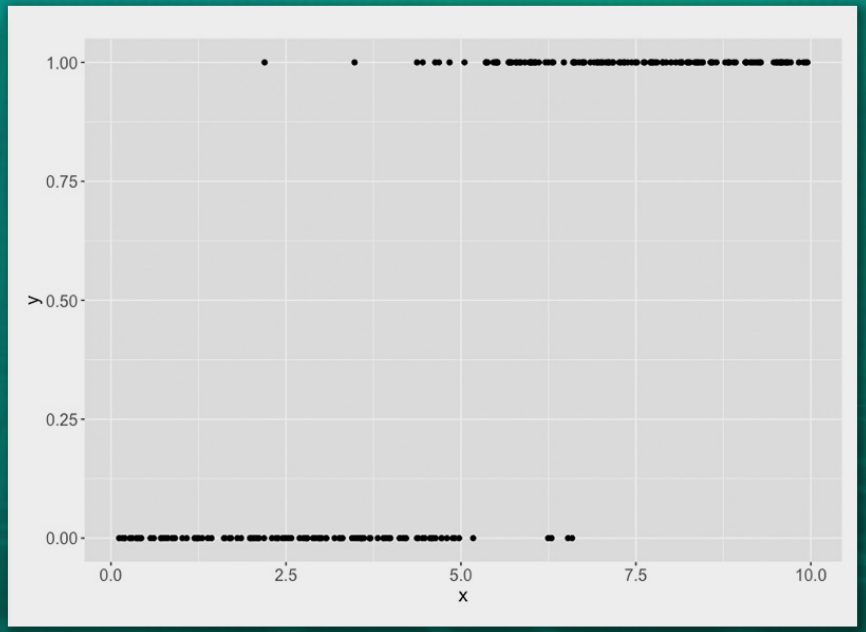

```
> ( (rank(x)-0.5) / length(x)  )  %>% qnorm()
```

```
# R ではrank で順番を返す
# これがx 座標
```

ロジスティック回帰(1)

例 放送授業を見た時間と合格したかどうか
見る時間が長いほど合格の確率は高くなるが高い確率でも失敗する

```
set.seed(100)
N <- 250
xmin <- 0; xmax<- 10
x<- runif(N,xmin,xmax)
a <- -10
b <- 2
p <-  exp(a+b*x) / ( 1+exp(a + b*x) )
y <- vector("numeric",length=N)
for(i in 1:N){
   y[i] <- rbinom(1,size=1,prob=p[i])
 }
train <- tibble(x=x,y=y)
```



ロジスティック回帰(2)

ロジット変換

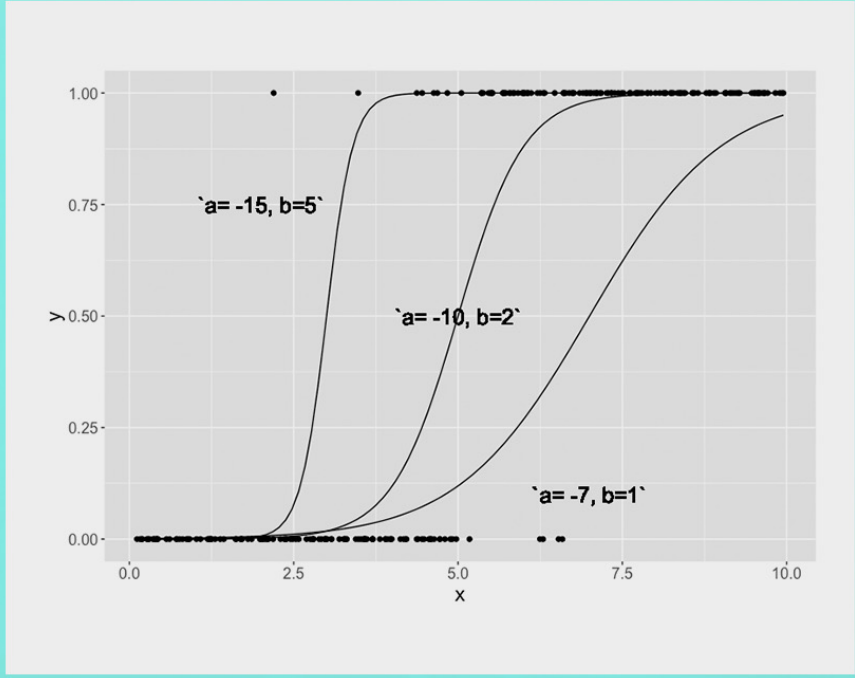
$\log\dfrac{p}{1-p}$ ：ロジット

$$\log\frac{p}{1-p} = a_0 + a_1 x_1 + \cdots + a_n x_n$$

$$\log\frac{p}{1-p} = a + bx \text{ とすると}$$

$$p = \frac{1}{1+\exp(-(a+bx))}$$

$$= \frac{\exp(a+bx)}{1+\exp(a+bx)}$$



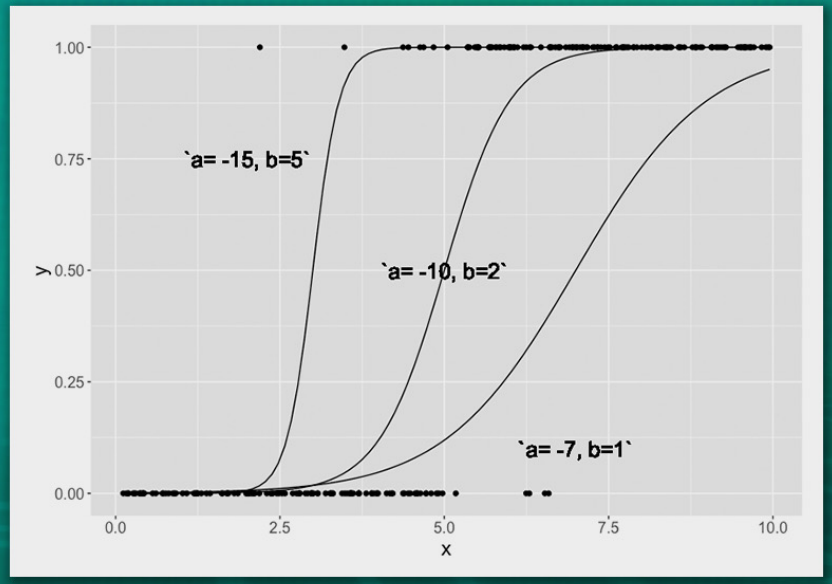
Rでの最尤(さいゆう)推定

```
b_lm1 <- glm(y~x,  data=train,  family=binomial)
b_lm1
```

family で分布を指定する

```
M <- 200
test <- tibble(
  x = seq(xmin,  xmax,  (xmax-xmin)  /  (M+1) ) )
y <- predict(b_lm1,
      newdata=test, type="response" )
test <- test %>% mutate(y=y)

ggplot(data=train,aes(x=x,y=y))+geom_point()+ +
geom_point(data=test,aes(x=x,y=y),size=0.1)
```



ルールの抽出

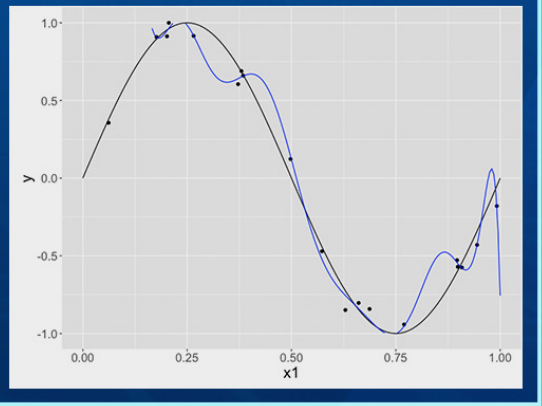
- 電話帳からいくつかの電話番号と名前を覚える

  ➡ 学習していない人の電話番号はわからない
  　電話番号にルールがない

- 気温や湿度から海の家の売上を予測する
- 過去数日の天気から次の日の天気を予測する

  ➡ 何らかのルールに従って
  　変動しているものに対して用いる

- 背後にある確率分布を想定する
- 多項式　三角関数などで近似する
- 機械学習



サンプルの抽出(1)

- データを元にその中にあるルールを抽出する
- データを訓練用のデータと検証用のデータに分けることがある
- (データの順番が結果に影響する時など)順番をシャッフルする
- sample
- 復元抽出と非復元抽出

```
sample(1:6,3)                  # 1から6の中から3個
sample(1:6,3,replace=T)        # 復元抽出
```

サンプルの抽出(2)

- iris あやめ(花)のデータ(データのサイズは150個)
- 3種類のあやめ　がく片の幅と花びらの長さ
- 花の形で種類を分別できるかどうか
  例 150個のものを120個と30個に分ける

```
x  <-  sample(1:150,120)
x
data1 <- iris[x,]
data2 <- iris[-x,]
```

```
# 行番号が x のものだけ抽出
# 行番号が x にないものだけ抽出
```