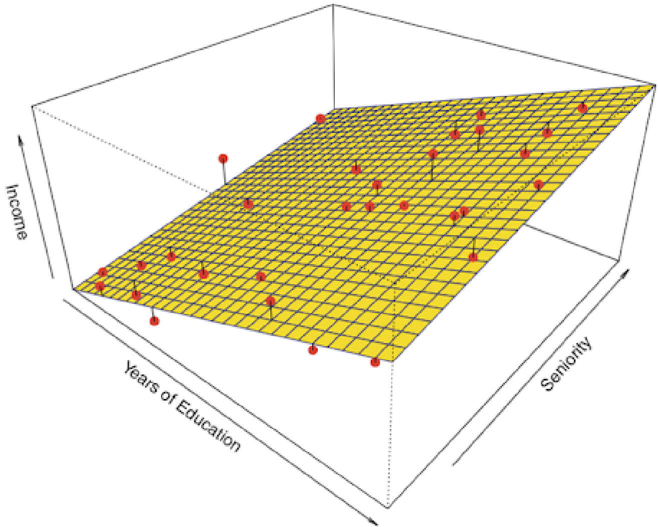
Introduction to data science & artificial intelligence (IF7100)

Arthur Charpentier

#322 Multiple Regression

été 2020

Linear Regression



From 1 to k features

$$y_i = \beta_0 + \beta_1 x_{i,1} + \cdots + \beta_k x_{i,k} + \varepsilon_i \text{ or } y_i = \boldsymbol{x}_i^\top \boldsymbol{\beta} + \varepsilon_i,$$

From a mathematical perspective, use matrix notations

$$\underbrace{\begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}}_{\boldsymbol{y}, n\times 1} = \underbrace{\begin{pmatrix} 1 & x_{1,1} & \cdots & x_{1,k} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n,1} & \cdots & x_{n,k} \end{pmatrix}}_{\boldsymbol{X}, n\times(k+1)} \underbrace{\begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{pmatrix}}_{\boldsymbol{\beta}, (k+1)\times 1} + \underbrace{\begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}}_{\varepsilon, n\times 1}.$$

i.e. $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$. Assume that $\mathbb{E}(\varepsilon_i) = 0$ and $\text{Var}[\varepsilon_i] = \sigma^2$.
The OLS estimator is $\widehat{\boldsymbol{\beta}} \in \text{argmin} \sum_{i=1}^n \left(y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}\right)^2$, i.e.

$$\widehat{\boldsymbol{\beta}} = \text{argmin}\big(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\big)^\top \big(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\big)$$

$$\widehat{\boldsymbol{\beta}} = \big(\boldsymbol{X}^\top \boldsymbol{X}\big)^{-1} \boldsymbol{X}^\top \boldsymbol{y}$$

Linear Regression

Then $\mathbb{E}(\widehat{\boldsymbol{\beta}})=\boldsymbol{\beta}$ and $\text{Var}(\widehat{\boldsymbol{\beta}})=\sigma^2\big(\boldsymbol{X}^\top\boldsymbol{X}\big)^{-1}$.
Assuming either that $\varepsilon_i\sim\mathcal{N}(0,\sigma^2)$ or n large,

$$\widehat{\boldsymbol{\beta}}\approx\mathcal{N}(\boldsymbol{\beta},\sigma^2\big(\boldsymbol{X}^\top\boldsymbol{X}\big)^{-1})$$

As previously (#321) we can test for significance $H_0:\beta_j=0$ and we can derive some confidence intervals, for β_j (for any j)

For prediction, $\widehat{y}=\widehat{\beta}_0+\widehat{\beta}_1x_1+\cdots+\widehat{\beta}_kx_k$.

Ceteris paribus can be translated into "all other things being equal" or "holding other factors constant"
Mutatis mutandis approximately translates as "allowing other things to change accordingly" or "the necessary changes having been made"

Regression

```
> import numpy as np
> import statsmodels.api as sm
> x = np.array([[5,1], [15,4], [25,-5], [35,4],
    [45,-2], [55,2]])
> x = sm.add_constant(x)
> y = np.array([5, 20, 14, 32, 22, 38])
> model = sm.OLS(y, x)
> results = model.fit()
> print(results.summary())
=====================================================
          coef.  std err       t   P>|t|    [0.025   0.975]
-----------------------------------------------------
const     4.0581   3.370   1.204   0.315    -6.668   14.785
x1        0.5578   0.097   5.770   0.010     0.250    0.865
x2        1.5604   0.508   3.071   0.055    -0.057    3.178
-----------------------------------------------------
Dep. Variable:    y        R-squared:         0.931
Model:            OLS      Adj. R-squared:    0.886
                           F-statistic:       20.37
```

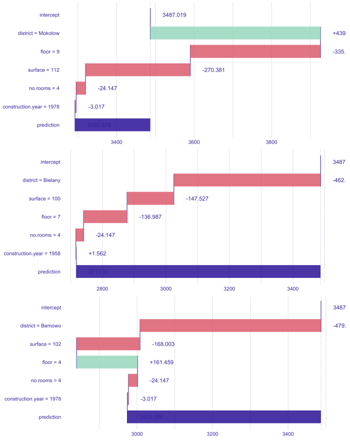
Regression

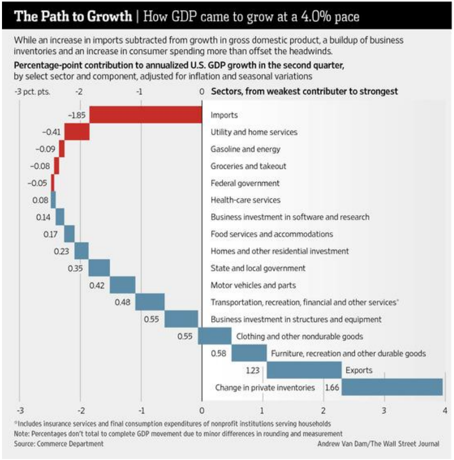
```
> df = data.frame(x1 = c(5, 15, 25, 35, 45, 55),
                  x2 = c(1, 4,-5, 4,-2, 2),
                  y  = c(5, 20, 14, 32, 22, 38))
> model = lm(y~x1+x2, data=df)
> summary(model)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.05810    3.37049   1.204   0.3149
x1           0.55783    0.09667   5.770   0.0103 *
x2           1.56037    0.50818   3.071   0.0545 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.'

Residual standard error: 4.037 on 3 degrees of freedom
Multiple R-squared: 0.9314, Adjusted R-squared: 0.8857
F-statistic: 20.37 on 2 and 3 DF,  p-value: 0.01796
```

Break-down for Additive Models





See The Wall Street Journal, on accounting equations

Partial Dependence Plot

Introduced in Greedy function approximation: A gradient boosting machine, Friedman (2001)

Let $\boldsymbol{x}$ be splited in two parts : $\boldsymbol{x}_s$ (variable(s) of interest) and $\boldsymbol{x}_c$ the complementary, $\boldsymbol{x} = (\boldsymbol{x}_s, \boldsymbol{x}_c)$. Partial dependence of $\boldsymbol{x}_s$ is

$$p(\boldsymbol{x}_s) = \mathbb{E}\big[m(\boldsymbol{x}_s, \boldsymbol{S}_c)\big] \text{ and } \widehat{p}(\boldsymbol{x}_s) = \frac{1}{n}\sum_{i=1}^{n} \widehat{m}(\boldsymbol{x}_s, \boldsymbol{x}_{i,c})$$