REM Tutorial 3: Regression I - Hedonic

Prof. Dr. Kathleen Kürschner Rauck

AT 2023

Hedonic (House) Pricing

- Heterogeneous goods → buying a house always means selecting a whole bundle of housing characteristics

- No price per standard quantity or quality of housing available (observable market price is an outcome of the unobservable utility from house ownership and use)

- **Hedonic price equation:** observable market price is modeled as a function of observable housing characteristics

$$P = \alpha + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_n X_n \tag{1}$$

  with:
  P = market price
  α = constant term
  X_i = characteristic i ($i = 1, \dots, n$), e.g., size, age, neighbourhood
  β_i = **implicit increase in price** for an additional unit of characteristic i

- Various applications in empirical research (see, e.g., academic GIFt:
  https://www.youtube.com/watch?v=0_NreXsVvVY&t=12s
  Paper: https://link.springer.com/article/10.1007/s11116-022-10295-8)

Today's Agenda

Learning Outcomes	R Functions	Packages	Data
Be able to plot kernel density estimates	max(); density()		Lucas_County_data
	polygon()		
Know how to add a curve to a plot	curve(); dnorm()		
	mean(); sd()		
Be able to perform the Jarque-Bera test in R	jarque.bera.test()	tseries	
		quadprog	
Be able to run an OLS regression using R	lm()		
& assess regression output	summary()		
Be able to produce a two-way scatter plot	plot()		
& remove scientific notation from axes	options()		
& add a fitted regression line to the plot	abline()		
Understand/recap common functional forms			
Know how to include & interpret a quadratic	I()		
term in a linear regression model			
Be able to plot a non-linear fitted line	data.frame(); seq()		
& determine minimum/maximum point	min(); max()		
	nrow(); predict()		
	as.data.frame()		
	lines(); abs()		
Know how to interpret coefficient estimates	as.factor()		
for categorical variables	levels()		
Know how to produce contingency tables	table()		

Practice

Work through **Tutorial 3 - Workbook** to solve the practical problems provided in **Tutorial 3 - Problem Set**.

Follow up

I'm adding some code of what I showed in class on alternative ways to generate squared terms etc. (not contained in Workbook):

Alternatively, we could generate age squared manually and add this variable to regression equation:

```
dat1$age2 <- (dat1$age)^2 # gen. age squared and add to dat1
fit2.2 <- lm(price ~ age + age2, data = dat1)
summary(fit2.2) # same result
```

Or we use the poly function to generate age^2 age^3 etc. and write result to **dat1**:

```
age.poly <- poly(dat1$age, 3, raw = TRUE) # evaluates raw, not orthogonal, polynomials to the constant polynomial (degree=0)
head(age.poly) # to view: 1st col is age, 2nd is age^2, 3rd is age^3
# add to dat1 in one step
dat1$age3 <- poly(dat1$age, 3, raw = TRUE)[,3]
```