The Evaluation of Management Practices

Jesper Hansen, Dirk Sliwka

Fall Term 2020

Contents

0. (Introduction to Python)

1. Introduction to the course

2. Regression Essentials

3. Statistical Tests

4. Predictions and Machine Learning

5. Regression and Causality

6. Using Panel Data

1. Introduction

Key questions addressed in this lecture:

- How can we evaluate the effect of management practices on outcomes variables such as profit, turnover, or job satisfaction?

- Why and when are regressions useful?

- How can we identify causal effects?

- What is the underlying idea of these methods?

- How do we work with the data?

- How do we analyze cross-sectional and longitudinal data sets?

- How can a field experiment be set up?

Useful literature:

- Angrist and Pischke (2008) Chapters 2 and 3

- Angrist and Pischke (2014)

- Wooldridge (2003) (Background reading)

- James, Witten, Hastie, Tibshirani (2017): An Introduction to Statistical Learning with Applications in R

- Andrea Ichino's lecture slides (for some links to standard econometrics courses): http://www.andreaichino.it/teaching_material.html

Key distinction for study designs:

Study based on *observational* data	**Laboratory experiment**	**Field experiment**
• Data creation process not affected by the researcher	• Data generated by the researcher in the lab	• Also: RCT (Randomized Controlled Trial), or in practice A/B test
• Example data: Data from surveys, balance sheets, personnel records,…	• Typically students are hired to make certain decisions/work	• Data generated in the field (for instance in a firm)
• Typically no *exogenous variation* in management practices (i.e. differences in use of practices may be related to unobserved variables)	• Exogenous treatment variation allows to study causal effects	• Exogenous treatment variation allows to study causal effects

Types of Data

- When evaluating management practices it is often useful to combine different types of data

- Qualtrix/SAP for instance distinguish X- and O-data

"X-data" Experience data	"O-data" Operational data
• Essentially data from surveys • Perceptions and Attitudes • Examples: Employee satisfaction, Customer satisfaction,…	• Data from IT systems on operational processes • Examples: Quit rates, bonuses, salaries, sales, profits, hiring durations

2. Regression Essentials

Suppose we are interested in the connection between

- an outcome variable y (e.g. workers effort, job satisfaction,..)
- and a variable x which may affect y (e.g. wage, the size of bonus payments, whether the firm uses performance pay or not,...)

Let e be a variable which describes all other determinants of y that we do not observe

Then we can denote the relationship between y and x as

$$y = f(x, e) \tag{1}$$

Key aim: Understand this function and learn about it by analyzing data

Distinction: Prediction and Causality

(i) Prediction

- Question: to what extent does knowing x allow us to predict y?

- Example:
    - When we as observers see that a company uses performance pay
    - What can we predict about the job satisfaction of its employees?
    - In other words: Is employee satisfaction higher in firms that use performance pay?

(ii) Causality

- Question: to what extent does a change of x lead to a change of y?

- Example:
    - A firm introduced performance pay
    - We want to know how this affected employee satisfaction
    - In other words: Did the change in performance pay *cause* a change in employee satisfaction?

These are different questions!

Further examples:

- *Education and wages*

  The fact that more educated people earn more does not tell us that education causes higher earnings

- *Gender diversity and performance*

  The fact that successful firms employ more women on boards does not tell us that a higher share of women causes a higher performance

Note:

- Answering the first (prediction) is typically substantially simpler than answering the second (causality)

- In the public debate (and also still in some fields in academia) these questions are often confounded

- We will start by thinking about the first question and then move to the second

The key idea of the following:

- Question: Why are regressions so important in empirical research?

- Answer:

  - Because they provide useful approximations to *conditional expectation functions*

  - And *conditional expectation functions* are a powerful tool to predict outcomes

- But:

  Without further ingredients they do not automatically detect causal relationships

2.1 The Conditional Expectation Function

- Think of X_i and Y_i as random variables (where X_i may be a vector)

- We are interested in the *conditional expectation function* (CEF) of Y_i given X_i in the population

$$E[Y_i|X_i]$$

- Useful interpretation:

Think of $E[Y_i|X_i]$ as a function stating the mean of Y_i among all people who share the same value(s) of X_i

- If Y_i is discrete and takes values out of a set T

$$E[Y_i|X_i = x] = \sum_{t \in T} \Pr(Y_i = t|X_i = x) \cdot t$$

where $\Pr(Y_i = t|X_i = x)$ is the conditional probability that $Y_i = t$ when $X_i = x$

Distinguish:

- *Population:* Complete group of potential observations for our question (for example: all working age people living in Germany, all US firms...)
- A *sample*: the observations that we can use for our research
  - employees who take part in a survey study like the GSOEP or LPP
  - set of firms for which we have information on management practices
  - subjects taking part in an experiment
- We can estimate the population CEF from a representative sample
  - If we for instance observe pairs (Y_i, X_i) for $i = 1, \dots, n$
  - We can estimate the conditional expectation of Y_i for a specific value of $X_i = x$ by taking the average of Y_i across observations with $X_i = x$

$$\tilde{Y}(X = x) = \frac{1}{|\{i | X_i = x\}|} \sum_{\{i | X_i = x\}} Y_i$$

  - Note: $\{i | X_i = x\}$ is the set of all observations for which $X_i = x$ and $(|\{i | X_i = x\}|$ is the number of observations for which $X_i = x$

Example: The CEF of earnings as a function of years of education

Source: Angrist/Pischke (2008, p. 31)

Statistical Analyses using Python
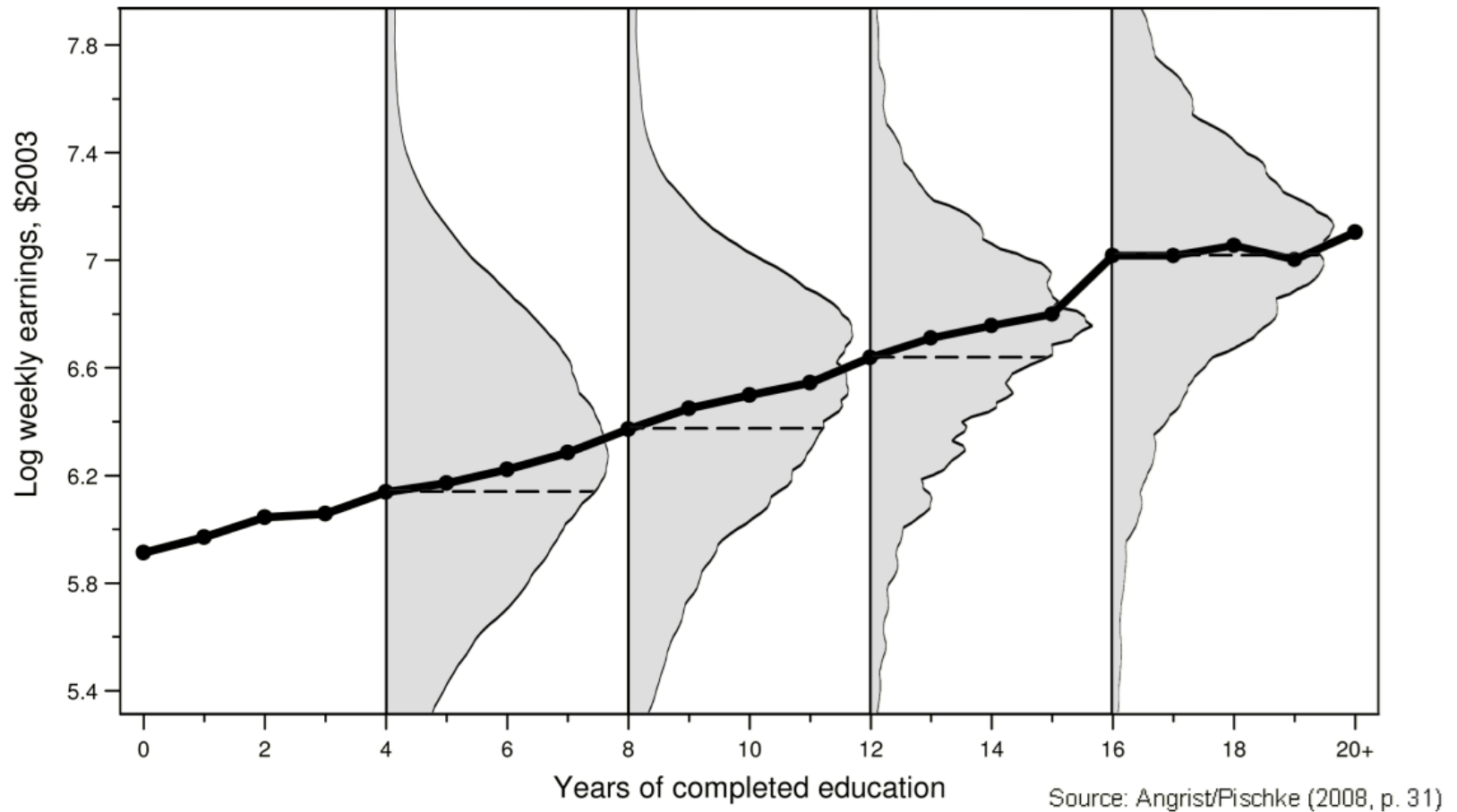
There are several packages/modules in Python that can be used to perform statistical analyses

- *NumPy* is the underlying package for scientific computing

- *Pandas*: provides data structures

- *Statsmodels*: to perform regressions

- *Seaborn:* to visualize data with graphs

- In the beginning of our Python file we import these modules

```
import pandas as pd
import numpy as np
import statsmodels.api as sm
import seaborn as sns
```

- We then call methods from these modules by something like

```
df=pd.read_csv(path_to_data)
```

(Here: call method read_cv from pandas)

Statistical Analyses using Python

Key concepts:

- *DataFrame* is is a 2-dimensional data structure
    - Provides by Pandas
    - Like an Excel spreadsheet
    - *Columns* contain variables (example: age, wage)
    - *Rows* contain observations (example: different people)
    - The first column contains an *index* (a label for the row)
    - On the previous slide: `df=pd.read_csv(path_to_data)` reads a table from the file and stored it in a new DataFrame called df
- *Series* is like a list containing one variable
    - Also has an *index*
- Behind all that are *numpy arrays* (*ndarrays*)
    - Grid of values of the same type

Printing Summary Statistics

- We typically start an analysis by looking at descriptive statistics

  - What are the means of the key variables?

  - What is their standard deviations?,…

- To print summary statistics use the `describe()` method

  - `df.describe()` prints summary statistics for all columns

  - `df['varname'].describe()` or `df.varname.describe()` prints summary statistics for variable `varname`

- We can also explore summary statistics for specific subgroups

- To print summary statistics by subgroup use

  `df.groupby('country').varname.describe()`

Graphs in Python

- It is often useful to visualize data with graphs
- Particularly useful: package *Seaborn* (`import seaborn as sns`)

Examples:

- `sns.barplot(x='country', y='income', data=df)`
  - Plots one bar for each realization of x with height equal to mean of y
  - Note: illustrates the estimated CEF for categorial variables
  - Adds confidence bands: Estimates areas where true population mean lies with 95% probability

- `sns.relplot(x='income', y='happiness', data=df)`
  - Plots scatter plot where each dot is a data point

- `sns.distplot(df['wage'])`

  - Plots histogram of the variable

  - Note `df['x']` returns a series of all observations of variable x

- Field experiment in a university library (Ockenfels/Sliwka/Werner (2015))

- Research question: Can performance be increased by splitting wage increases?

- Task: inserting adhesive labels with barcodes into the book stock (approximately 150,000 books) to enable automated borrowing procedures.

- Agency specialized in recruiting temporary workers hired workers

  - for a one-time job opportunity that would last for seven working hours

  - consisted of library inventory task for a fixed hourly wage

  - 99 people signed up for the job

- Three treatments:

  - Baseline: Fixed hourly wage of €8 (as announced in advertisment)

  - Gift_1: Hourly wage of €12 announced upon arrival

  - Gift_2: Hourly wage of €10 Euro & raised at half time to €14

Analyse data from a Field Experiment

- Let us analyse data from Ockenfels/Sliwka/Werner (2015)

- Please write Juypiter notebook

- First import modules

  - ```
    import pandas as pd
    ```
  - ```
    import numpy as np
    ```
  - ```
    import statsmodels.api as sm
    ```
  - ```
    import statsmodels.formula.api as smf
    ```
  - ```
    import seaborn as sns
    ```

- Read the data into a DataFrame

  - ```
    path_to_data='https://raw.githubusercontent.com/armou
    tihansen/EEMP2020/main/datasets/libraryExpData.csv'
    ```

  - ```
    df = pd.read_csv(path_to_data)
    ```

- Click on the DataFrame in the variable explorer and inspect the data set

Your Task   **Analyse data from a Field Experiment**

- Inspect the data (variable `tr` tells you to which treatment an observation belongs)

- Compare the mean of total performance between the three treatments
  - Note: To do this, it is convenient to use the groupby method
  - Syntax (adapt!): `df.groupby('country')['wage'].describe()`

- Visualize the treatment differences with a barplot
  (Adapt: `sns.barplot(x='country', y='income', data=df)`)

- Save your notebook as `analyzeLibEx` to extend it later

Two key results(for the proofs see Angrist/Pischke (2009, pp 32)

Result: CEF Decomposition Property

We can decompose Y_i such that $Y_i = E[Y_i|X_i] + \varepsilon_i$

(i) where ε_i is mean independent of X_i that is $E[\varepsilon_i|X_i] = 0$

(ii) and therefore ε_i is uncorrelated with any function of X_i

- Therefore: A random variable Y_i can be decomposed into a piece that is "explained by X_i" (the Conditional Expectation Function) and a piece that remains unexplained by any function of X_i

- In the example: We can decompose the wage of a person
  - in a piece that is "explained" by education (i.e. the CEF)
  - and piece that is left over
  - and this latter piece is uncorrelated ("orthogonal to") with any function of education

Result: CEF Prediction Property

Let $m(X_i)$ be any function of X_i. The CEF solves

$$E[Y_i|X_i] = \arg\min_{m(X_i)} E\left[\left(Y_i - m(X_i)\right)^2\right]$$

so it is the best predictor of Y_i given X_i in the sense that it solves the minimum mean square error (MMSE) prediction problem.

- The CEF is a very useful predictor: If I observe other related variables and „plug them into the CEF" the value of the CEF comes close to the true value of the outcome variable

- We want a function (call it $m(X_i)$) that gives us a good prediction for Y_i

$$\hat{Y}_i = m(X_i)$$

- Important criterion: The distance between $\hat{Y}_i$ and Y_i should be small

- The result now states: When we use the quadratic distance $\left(Y_i - m(X_i)\right)^2$, then the CEF is the best function we can find

Therefore:

- The CEF provides a natural summary of empirical relationships
  - It gives the population average of Y_i for the group of people having the same X_i
  - It describes the best (MMSE) predictor of Y_i given X_i
  - It allows to decompose Variance in the data (see Appendix)

- If I know the CEF I can make predictions which value Y_i would take for different values of X_i

  (Note: in the population; not in the sense of a causal change in Y_i because of a change of X_i!)

But: What is connection between the CEF and regression analysis and machine learning?

- In the following: regressions and other machine learning algorithms are tools to approximate the CEF

2.2 Regression and Conditional Expectations

- Typically we will not know the functional form of the CEF when Y is a continuous variable

- But we can try to approximate it

- Start with simple case of two variables and consider the linear function

$$Y_i = \beta_0 + \beta_1 X_i$$

- Now determine β_0 and β_1 such that

$$(\beta_0, \beta_1) = \arg \min_{b_0, b_1} E[(Y_i - b_0 - b_1 X_i)^2]$$

- Let us call this the *Population Regression Function (PRF)*

- Of all possible linear functions of X_i – which one gives us the least (quadratic) deviation from Y_i in expected terms?

$$(\beta_0, \beta_1) = \underset{b_0, b_1}{\operatorname{argmin}} E[(Y_i - b_0 - b_1 X_i)^2]$$

First order conditions

$$E[2(Y_i - b_0 - b_1 X_i)] = 0 \qquad\qquad (2)$$

$$E[2(Y_i - b_0 - b_1 X_i)X_i] = 0 \qquad\qquad (3)$$

Hence, from (2) and (3)

$$b_0 = E[Y_i] - b_1 E[X_i]$$

$$b_1 E[X_i^2] = E[X_i Y_i] - b_0 E[X_i]$$

such that

$$b_1 = \frac{E[Y_i X_i]}{E[X_i^2]} - (E[Y_i] - b_1 E[X_i])\frac{E[X_i]}{E[X_i^2]}$$

$$\Leftrightarrow b_1 = \frac{E[Y_i X_i] - E[Y_i]E[X_i]}{E[X_i^2] - (E[X_i])^2}$$

- Hence, in the bivariate case

$$\beta_1 = \frac{E[Y_i X_i] - E[Y_i]E[X_i]}{E[X_i^2] - (E[X_i])^2} = \frac{Cov[Y_i, X_i]}{V[X_i]} \qquad (4)$$

- This is the population version of OLS regression for the bivariate case
- Define: The population residual

$$e_i = Y_i - b_0 - b_1 X_i$$

- Note that $Cov[e, X_i] = E[eX_i] - E[e]E[X_i] = 0$
  - as $E[Y_i - b_0 - b_1 X_i] = 0$
  - and $E[(Y_i - b_0 - b_1 X_i)X_i] = 0$ (from the first order conditions)
- Hence, *the population residual is uncor*related with X_i
- We can proceed analogously in the multivariate case (where β and X_i are vectors)

$$\beta = E[X_i X_i']^{-1} E[X_i Y_i]$$

(See Econometrics textbook for details of the proof)

From a Sample to the Population

- So far we spoke about whole populations but in reality we (typically) do not know the population parameters

- We work with samples (subsets) of a population but we want to say something about the population

- That is we want to estimate the population parameters β using a sample

- And we want to have an idea how good these estimates are

- The aim is therefore to estimate the population parameters β from a sample

We want to

- obtain the estimated coefficients $\hat{\beta}$

- and learn about the precision of these estimates

The Bivariate Case: We want to estimate the parameter $\beta_1 = \frac{Cov[Y_i, X_i]}{V[X_i]}$

- We have a sample of size N and thus observe (Y_i, X_i) for $i = 1, \ldots N$

- We can estimate

  - $Cov[Y_i, X_i]$ by the sample covariance $\frac{1}{N}\sum_{i=1}^{N}(X_i - \bar{X})(Y_i - \bar{Y})$

  - $V[X_i]$ by the sample variance $\frac{1}{N}\sum_{i=1}^{N}(X_i - \bar{X})^2$

- And this leads to the OLS *estimator* $\hat{\beta} = \frac{\frac{1}{N}\sum_{i=1}^{N}(X_i - \bar{X})(Y_i - \bar{Y})}{\frac{1}{N}\sum_{i=1}^{N}(X_i - \bar{X})^2}$

Multivariate Case: We want to estimate $\beta = E[X_i X_i']^{-1} E[X_i Y_i]$

- We observe (Y_i, X_i') for $i = 1, \ldots N$, that is

  - $(Y_1, X_{10}, X_{11}, X_{12}, \ldots X_{1K-1})$,

  - $(Y_2, X_{20}, X_{21}, X_{22}, \ldots X_{2K-1})$,...

- We can estimate $E[X_i X_i']$ by $\frac{1}{N}\sum_{i=1}^{N} X_i X_i'$ and $E[X_i Y_i]$ by $\frac{1}{N}\sum_{i=1}^{N} X_i Y_i$

- And this leads to the OLS *estimator* $\hat{\beta} = \left[\sum_{i=1}^{N} X_i X_i'\right]^{-1} \sum_{i=1}^{N} X_i Y_i$

OLS Regressions in Python

- We can use the module statsmodels & it is convenient to use "formulas"

- Import: `import statsmodels.formula.api as smf`

- If you have a DataFrame df containing variables y, x1 and x2 and you want to regress y (dependent variable) on x1 and x2 (indep. variables):
  `reg = smf.ols('y ~ x1 + x2', data=df).fit()`

- And show the results with
  `print(reg.summary())`

- Note: one can also directly get nice regression tables (as reported in research papers) with different specifications with `summary_col`
  `from statsmodels.iolib.summary2 import summary_col`

- Example:

  - `reg1 = smf.ols('y ~ x1', data=df).fit()`
  - `reg2 = smf.ols('y ~ x1 + x2', data=df).fit()`
  - `print(summary_col([reg1, reg2],stars=True))`

Observational Data: Management Practices and Performance

Bloom und Van Reenen (2007), Bloom and Van Reenen (2012) study survey data

- Evaluate whether differences in the use management practices can explain productivity differences between firms

- Use an interview-based evaluation tool to assess 18 basic management practices

- Run the survey in many industries and countries

- Interviewers give a score from 1-5 on the 18 practices

- Compute a management score computed from the surveys

- Study the association between

  - the management score and

  - the financial success of the companies (e.g. sales, ROCE)

Management Practice Dimensions
(examples, see Bloom und Van Reenen (2010) , p. 206)

- *Introduction of modern manufacturing techniques*
  What aspects of manufacturing have been formally introduced [...]?

- *Rationale for introduction of modern manufacturing techniques*
  Were modern manufacturing techniques adopted just because others were using them, or are they linked to meeting business objectives like reducing costs and improving quality?

- *Performance tracking*
  Is tracking ad hoc and incomplete, or is performance continually tracked and communicated to all staff?

- *Performance dialogue*
  In review/performance conversations, to what extent is the purpose, data, agenda, and follow-up steps (like coaching) clear to all parties?

- *Consequence management*
  To what extent does failure to achieve agreed objectives carry consequences, which can include retraining or reassignment to other jobs?

- *Target time horizon*
  Does top management focus mainly on the short term, or does it visualize short-term targets as a staircase toward the main focus on long-term goals?

- *Targets are stretching*
  Are goals too easy to achieve, especially for some sacred cows areas of the firm, or are goals demanding but attainable for all parts of the firm?

- *Managing human capital*
  To what extent are senior managers evaluated and held accountable for attracting, retaining, and developing talent throughout the organization?

- *Promoting high performers*
  Are people promoted mainly on the basis of tenure, or does the firm actively identify, develop, and promote its top performers?

- *Attracting human capital*
  Do competitors offer stronger reasons for talented people to join their companies, or does a firm provide a wide range of reasons to encourage talented people to join?

- Use data from Bloom, Genakos, Sadun and Van Reenen. "Management Practices Across Firms and Countries." The Academy of Management Perspectives, 26, no. 1 (2012): 12-33.

- Start a new Juypiter Notebook (you can copy the first part with the imports and adapt from the previous exercise, but save it under a different name)

- Read the data into a DataFrame

  - ```
    path_to_data =
    'https://raw.githubusercontent.com/armoutihansen/
    EEMP2020/main/datasets/AMP_Data.csv'
    ```

  - ```
    df = pd.read_csv(path_to_data)
    ```

- The data set for instance contains variables `management` (the management score across practices) and financial KPI `roce` (=EBIT/Capital employed)

- Type `df` to show the DataFrame

- Inspect the data set

- Inspect the data in more detail by plotting graphs, for instance use
  - `sns.displot(df.xvar)` to plot a histogram of a variable xvar
  - `sns.relplot(x='xvar', y='yvar', data=df)` for a scatter plot
- Now run a regression of `roce` as dependent variable on `management`
  - Recall the syntax (adapt!):
  - `reg = smf.ols('yvar~xvar1+xvar2', data=df).fit()` `print(reg.summary())`
- Interpret your result
- Save your notebook as `ManagementPractices` to reuse it later

2.3 Dummy Variables

When X_i is a single dummy variable that only takes value 0 or 1

- Then $E[Y_i|X_i = 0]$ is a constant and $E[Y_i|X_i = 1]$ is another constant and the CEF is fully characterized by these constants:

$$E[Y_i|X_i] = \underbrace{E[Y_i|X_i = 0]}_{\beta_0} + X_i \cdot \underbrace{(E[Y_i|X_i = 1] - E[Y_i|X_i = 0])}_{\beta_1}$$

is a linear function of X_i

- When I have precise estimates of the PRF then I have a precise estimate of $E[Y_i|X_i]$

Note:

- The PRF exactly describes the CEF

- Linearity is not an assumption but a fact

- This is a very common data structure for instance in an experiment: X_i indicates whether somebody is in the treatment instead of the control group

Generating New Variables

- New variables can be created by `df['newvarname']=`…

- You can also generate new variables and compute their value as a function of existing variables:

  `df['salesPerEmp']=df['sales']/df['emp']`

- A Boolean variable takes values *True* or *False*

  – A condition such as `(x>5)` gives back the value `True` when its true and otherwise `False`

- A Boolean variable can be used as a dummy variable

- A dummy variable can thus be created using a condition

  – Hence, `df['dummy']=(df['X']==5)` creates a dummy variable (column) that takes value `True` if the variable X is equal to 5

Your Task   **Analyse data from a Field Experiment**

- Open `analyzeLibEx` in which you analyzed the library experiment

- Compare again the mean of total performance between the three treatments

- Generate two new dummy variables for treatments 2 and 3
  - For instance use `df['dummyTr2']=(df['tr']==2)`

- Now add a regression of total_performance on the two treatment dummies

- Compare the regression results with the means from the summary statistics

- Save the file

2.4 Interaction Terms

- Sometimes we expect that the conditional expectation function $E[Y_i | X_{i1}, X_{i2}]$ is not additively separable such that it can sensibly be approximated by a population regression $Y_i = \alpha + \beta_1 X_{i1} + \beta_2 X_{i2}$

- But we may want to allow for the possibility that the effect of X_{i1} depends on the size of X_{i2}, for instance

  – The effect of performance pay on job satisfaction may depend on gender

  – The effect of a training may depend on experience,…

- In experiments we might consider a setting in which X_{i1} is a treatment dummy and X_{i2} is a specific characteristic of a treated object and we may want to study *heterogenous treatment effects*

- For instance the object is a

  – person and the characteristic is the age, gender, or experience.

  – firm and the characteristic is the size, industry, region,..

- When expecting that the effect of X_{i1} depends on the size of X_{i2} researchers typically estimate a regression

$$Y_i = \alpha + \beta_1 \cdot X_{i1} + \beta_2 \cdot X_{i2} + \beta_3 \cdot X_{i1} \cdot X_{i2} + \varepsilon_i$$

- We thus include an *interaction term* and approximate the CEF by a linear function from $\mathbb{R}^2 \rightarrow \mathbb{R}$

- Note: Never forget to include both variables as well as their interaction

- If we estimate a regression of this form the effect of X_{i1} on Y_i is approximately

$$\frac{\partial E[Y_i | X_{i1}, X_{i2}]}{\partial X_{i1}} \approx \beta_1 + \beta_3 \cdot X_{i2}$$

- β_3 thus estimates the extent to which the effect of X_{i1} depends on X_{i2}

Paying Gig Workers

- RCT with online fashion retailer Zalando (Butschek/Kampkötter/Sliwka 2019)

- Platform Zalon, where customers get curated shopping service

- Platform matches cutomers to „stylists" (freelancers) who recommend outfits

- How should these stylists be paid?
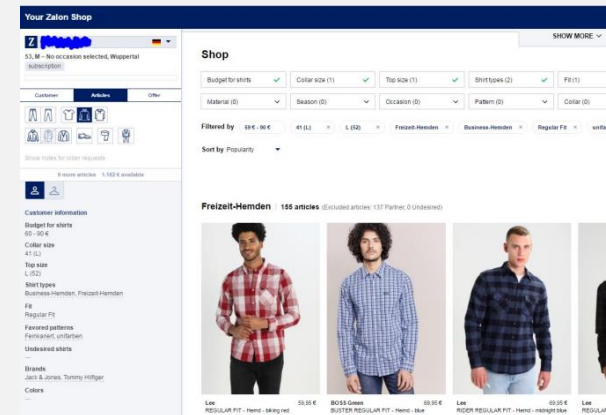
Treatments: New Stylists randomly assigned for first two month

- Control: Stylists paid on commission rate (%-share of sales)

- Treatment: Stylists receive fixed payment per customer & lower commission rate

Survey before the start:

- Stylist's risk preferences

- Motivation for the job

Paying Gig Workers

Table 2: Treatment effect on labor supply

	(1)	(2)
Treated	1.39	6.42
	(16.682)	(28.492)
Treated × risk averse		-25.65
		(36.391)
Risk averse		-1.71
		(17.243)
Adjusted R-squared	0.056	0.038
Number of observations	202	187

Note: Heteroskedasticity-robust standard errors in parentheses. * $p<0.10$, ** $p<0.05$, *** $p<0.01$. Outcome variable: normalized stylist-level total number of desired slots. Controls: randomisation stratum, hire month and treatment duration. Risk averse is a median split dummy from the baseline survey measure (1 item). F-test treated + treated*risk-averse: $p = 0.334$.

Paying Gig Workers

Table 3: Treatment effect on sales performance

	(1)	(2)
Treated	-4.43	-17.14***
	(3.946)	(5.686)
Treated × intrinsically motivated		26.70***
		(8.121)
Intrinsically motivated		-20.57***
		(5.747)
Adjusted R-squared	0.004	0.007
Number of observations	10,090	9,524
Number of stylists	202	187

Note: Standard errors clustered at stylist level in parentheses. * p<0.10, ** p<0.05, *** p<0.01. Outcome variable: normalized order-level sales. Controls: randomisation stratum, calendar week and potential experience (3rd-degree polynomial). Intrinsic motivation (2-item index) is a median split dummy from the baseline survey measure. F-test treated + treated*intrinsically motivated: p = 0.099.

Paying Gig Workers

Selecting Subsets of the Data
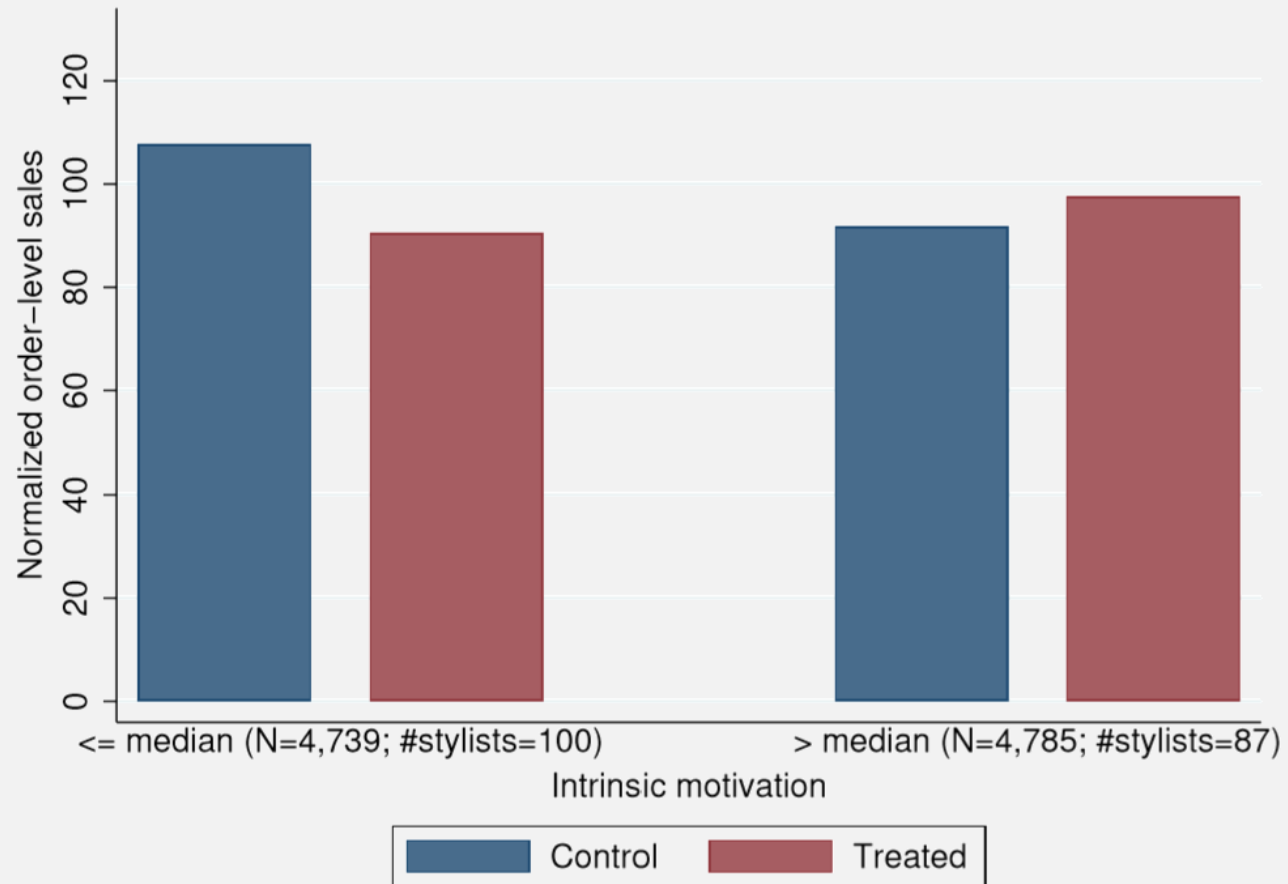
- Sometimes we want to use only a subset of the DataFrame, for instance if we want to run a regression only on a subset of the data

- Pandas has different methods for subset selection

- For instance, one could use the *indexing operator []* to select columns
  - `df['age']` gives back a series that contains only column age
  - `df[['age','wage']]` gives a DataFrame including only columns age & wage from the initial DataFrame df

- If we put a condition in the brackets, then rows are selected that satisfy this condition
  - `df[df['age']>50]` gives back a DataFrame containing only rows (observations) where age is larger than 50
  - We can use & (for and) and | (for or):
  - `df[(df['age']>50) | (df['age']<30)]` gives back a DataFrame that contains only observations where age<30 or >50

Categorial Variables and Interaction Terms in Regressions

- For categorial variables statsmodels formulae can automatically generate dummy variables for each category with the `C()` operator:

    ```
    smf.ols('Wage ~ age + C(Region)', data=df).fit()
    ```

- Interaction terms can also be directly generated with `*`

    ```
    smf.ols('Wage ~ age * female', data=df).fit()
    ```

- Note: when using * statsmodels also includes the two interacted variables separately

- Furthermore: You can use functions (from numpy) to transform variables directly in the regression equation

```
smf.ols('np.log(Wage) ~ age * female', data=df).fit()
```

- Note: the function `np.log(x)` computes the log of `x`

- Open your ManagementPractices.py file

- Research question: Is a management practice scoring that has been developed in one countries is equally predictive for performance in a country with a different culture?

- Background: the B/vR scoring has been developed in the UK

- Your task: Find out whether the management score is equally predictive for ROCE in China as compared to the UK

- First create a dummy variable `ChinaD` that includes only observations from China (inspect variable `country`)

- Then create a data frame that only includes data from the UK and China:
`dfn=df[(df["country"]=='China')|(df["country"]=='Great Britain')]`

- Now rerun your regression of `ROCE` on management interacting `management` with `ChinaD` (do not forget to run it on the dfn DataFrame!)

- Interpret your results

2.5 Estimating Non-linear functions

- In some applications we have reason to believe that the CEF is non-linear

- For instance, wages may first increase in age and then decrease

- Many applied researchers then start by estimating a quadratic function

$$Y_i = \alpha + \beta_1 \cdot X_i + \beta_2 \cdot X_i^2 + \varepsilon_i$$

- Hence, we approximate the CEF with a quadratic function

- This can also be useful when we suspect that the CEF is concave or convex

- But be careful when interpreting β_1: this is no longer the slope parameter but

$$\frac{\partial E[Y_i|X_i]}{\partial X_i} \approx \beta_1 + \beta_2 \cdot 2X_i$$

- Sign of β_2 estimates the second derivative of the function, as

$$\frac{\partial^2 E[Y_i|X_i]}{\partial X_i^2} \approx 2\beta_2$$

- Sometimes researchers replace the dependent variable with its logarithm

$$ln\, Y_i = \alpha + \beta \cdot X_i + \varepsilon_i$$

- Part of reason: Logs less sensitive to outliers & may reduce heteroscedasticity ($\rightarrow$ statistical tests)

- But more importantly: logs sometimes lead to convenient interpretations

- When X_i is a dummy variable our CEF is fully captured by a regression &

  - $ln\, Y_{i1} = \alpha + \beta + \epsilon_i$

  - $ln\, Y_{i0} = \alpha + \epsilon_i$

- Then $\qquad\qquad\qquad \beta = ln\, Y_{i1} - ln\, Y_{i0} = ln\frac{Y_{i1}}{Y_{i0}}$

- Such that $\qquad\qquad \frac{Y_{i1}}{Y_{i0}} = \exp(\beta) \approx 1 + \beta$

$\rightarrow$ The coefficent β is approximately equal to the percentage change in the outcome variable (approximation is ok for small enough β (like $\beta < 0.2$))

$\rightarrow$ The outcome is unaffected by the units in which Y_i is measured

Study | **Education and Wages**

- One of the classical problems in labor and personnel economics:

  What is the association between education and wages?

- Here: use the NLSY97, a nationally representative US sample of approximately 9,000 youths who were 12-16 years old in 1996

- Regress wages in 2012 on dummy variables for educational degrees

  - First use absolute wage levels (in $ amounts)

  - Then use the log of wages

- Note: We will discuss later on to what extent these regressions may capture causal effects

Hence:

- Regression provides the best linear predictor for the dependent variable; the CEF provides the best unrestricted predictor

- Even if the CEF is non-linear, regressions provide the best linear approximation

- A/P: This *"lines up with our view of empirical work as an effort to describe essential features of statistical relationships without necessarily trying to pin them down exactly"*

- Furthermore

  – Imposing linearity reduces complexity

  – A linear function is summarized in a few parameters that often have accessible interpretations

- But: there is danger of oversimplification

  – Other machine learning techniques allow to relax assumption of linearity or on specific functional forms

  – May allow to come closer to the true CEF in complex data