3. Statistical Tests

- So far, we only discussed how to obtain consistent estimates of a parameter and how to interpret regression estimates
- An important part of empirical work is to test whether the estimated parameters differ from a hypothesized quantity
- We obtain the estimate of $\hat{\beta}$ from a sample
- To do a statistical test we need to know something about the distribution of $\hat{\beta}$
- Useful thought experiment: think of the variance in the obtained estimates $\hat{\beta}$ when you would draw different samples from a population
  - When the variance of $\hat{\beta}$ is small then you can be very certain that $\hat{\beta}$ is close to the true β
  - When the variance of $\hat{\beta}$ is large then it may be far away from β and you probably don't learn so much from $\hat{\beta}$

3.1 Testing Hypotheses about a Parameter

Consider again the bivariate case where

$$\widehat{\beta_1} = \frac{\frac{1}{N}\sum_{i=1}^{N}(X_i - \bar{X})\,(Y_i - \bar{Y})}{\frac{1}{N}\sum_{i=1}^{N}(X_i - \bar{X})^2}$$

This can be rewritten (using that $Y_i = \beta_0 + \beta_1 X_i + e_i$) to become

$$\hat{\beta} = \beta_1 + \frac{\sum_{i=1}^{N}(X_i - \bar{X})\, e_i}{\sum_{i=1}^{N}(X_i - \bar{X})^2}$$

→ The estimate is the sum of the population value and a function of the residuals

When the residuals are independent and follow $N\left(0, \sigma^2\right)$ the variance of $\hat{\beta}$ is

$$V\,[\hat{\beta}\,|\,X_1, X_2, \ldots, X_N] = \frac{\sigma^2}{\sum_{i=1}^{N}(X_i - \bar{X})^2}$$

→ The estimated $\hat{\beta}$ fluctuate around the true β with variance $\frac{\sigma^2}{\sum_{i=1}^{N}(X_i - \bar{X})^2}$

We can use this to construct a test statistics for a coefficient

- Our Null hypothesis is $H_0: \beta = 0$
- The alternative hypothesis is $H_1: \beta \neq 0$

Note that as $\hat{\beta} \sim N\left(\beta, V\left(\hat{\beta}\right)\right)$ we have that

$$\frac{\hat{\beta} - \beta}{sd\left(\hat{\beta}\right)} \sim N(0,1)$$

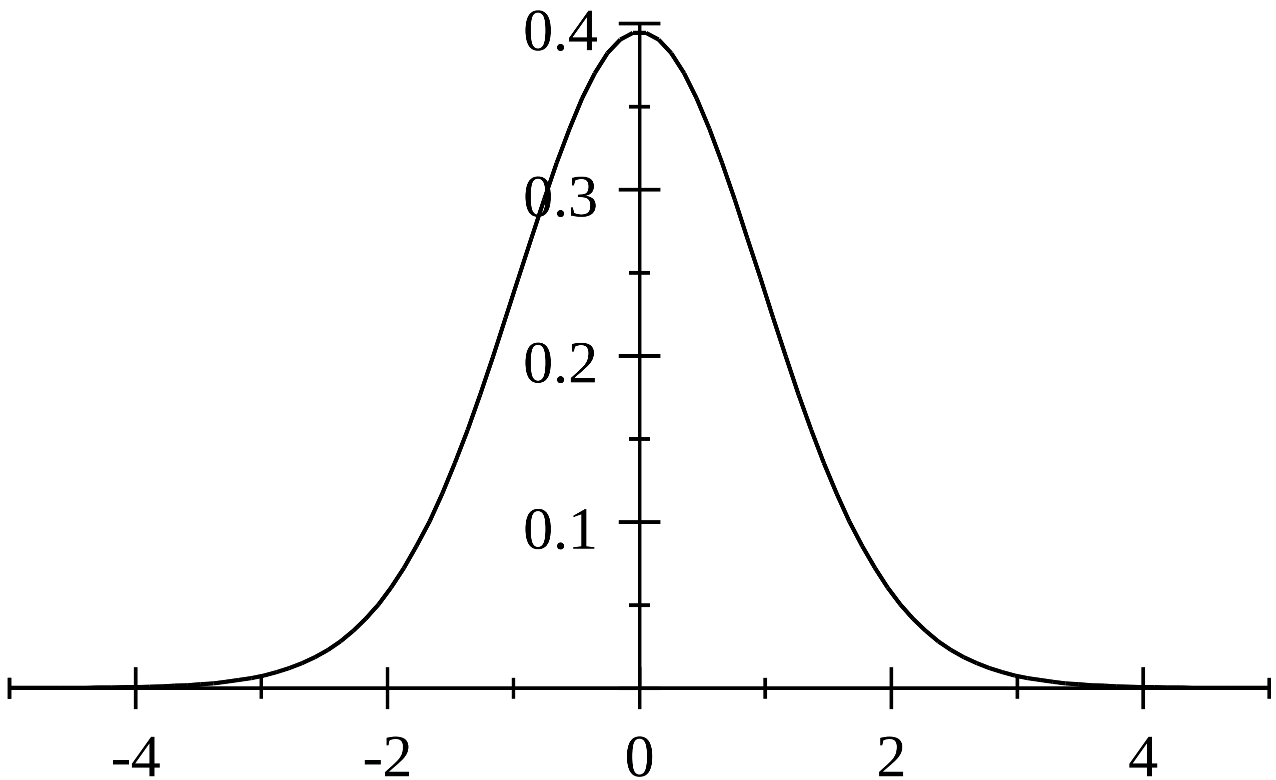
- As we do not know σ^2 and thus $sd\left(\hat{\beta}\right)$ we have to estimate it from the data
- This estimate is the standard error $se\left(\hat{\beta}\right)$ – see an econometrics textbook
- Note: $se\left(\hat{\beta}\right)$ is itself a random variable as it is an estimate based on the sample, but one can show:

  $\frac{\hat{\beta}-\beta}{se(\hat{\beta})}$ follows a Student‘s t-distribution with n-2 degrees of freedom:

$$\frac{\hat{\beta} - \beta}{se\left(\hat{\beta}\right)} \sim t(N-2)$$

Note: The t-distribution is close to the standard normal distribution

Example: Density of $t(25)$



The Multivariate Case

- One can show analogously that

$$\hat{\beta} = \beta + \left[\sum_{i=1}^{N} X_i \, X_i'\right]^{-1} \sum_{i=1}^{N} X_i \, e_i$$

- In matrix notation

$$\hat{\beta} = \beta + (X'X)^{-1}X'e$$

- When the residuals are normally distributed & have the same variance

$$e \sim N\left(0, \sigma^2 I_N\right)$$

where I_N is the $N \times N$ identity matrix

- One can show that the vector of parameter estimates

$$\hat{\beta} \sim N\left(\beta, \sigma^2 (X'X)^{-1}\right)$$

- And (if there are k parameters to estimate)

$$\frac{\hat{\beta} - \beta_j}{se\left(\hat{\beta}\right)} \sim t(N - k - 1)$$

Hence, to test the Null hypothesis that $\beta_j = 0$ we can look at the $t{-}statistic$

$$t_{\hat{\beta}} = \frac{\hat{\beta}_j}{se(\hat{\beta}_j)}$$

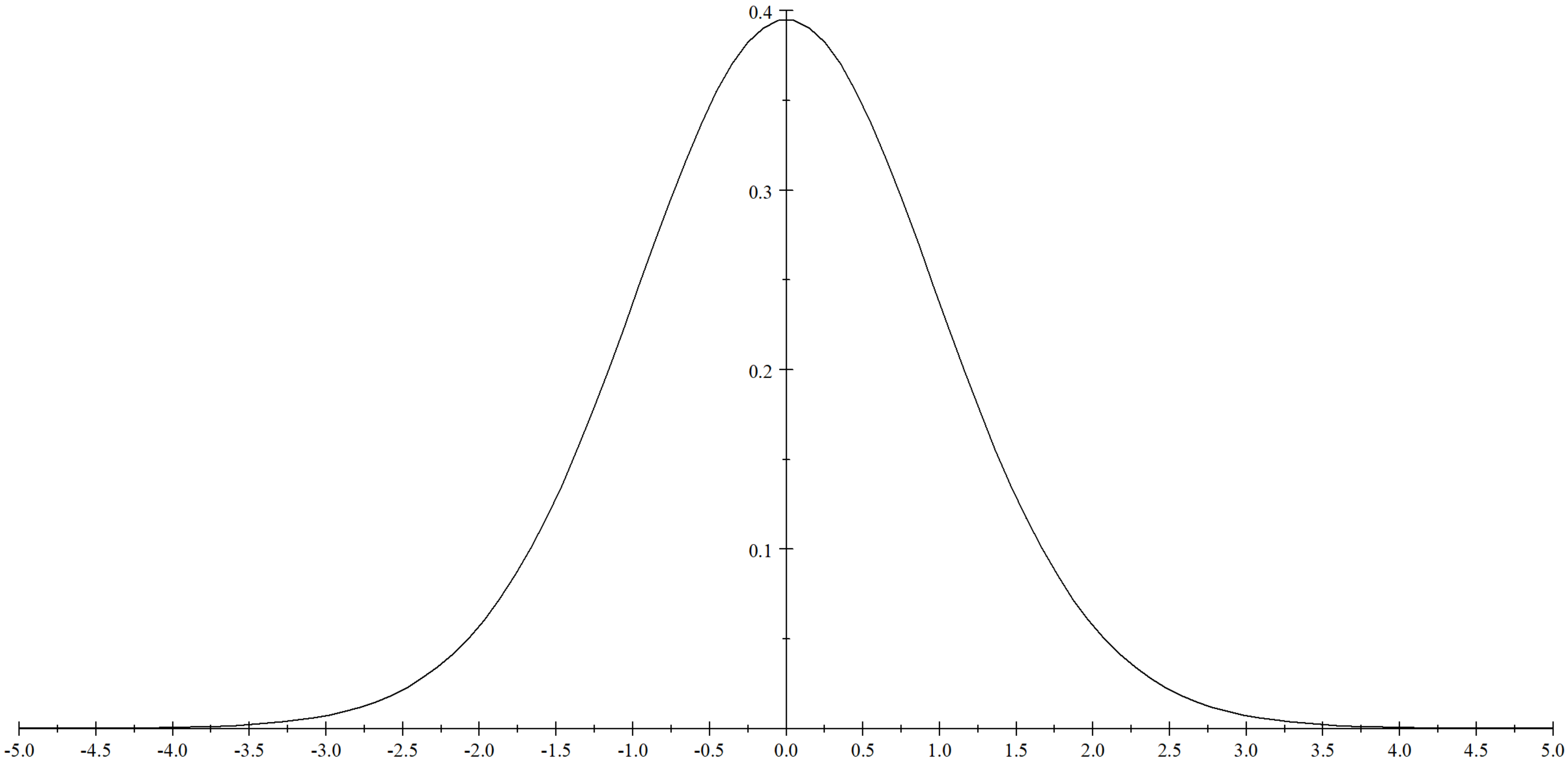
- When we consider a two-sided test,
  - we will reject H_0 whenever $t_{\hat{\beta}}$ is too large or too small
  - then it is unlikely that we would obtain an estimate $\hat{\beta}$ if the true β were 0
- Significance level α: likelihood that H_0 is rejected when it is in fact true
- Hence, we will reject H_0 at a significance level α if

$$\left|t_{\hat{\beta}}\right| > t_{\frac{\alpha}{2}}$$

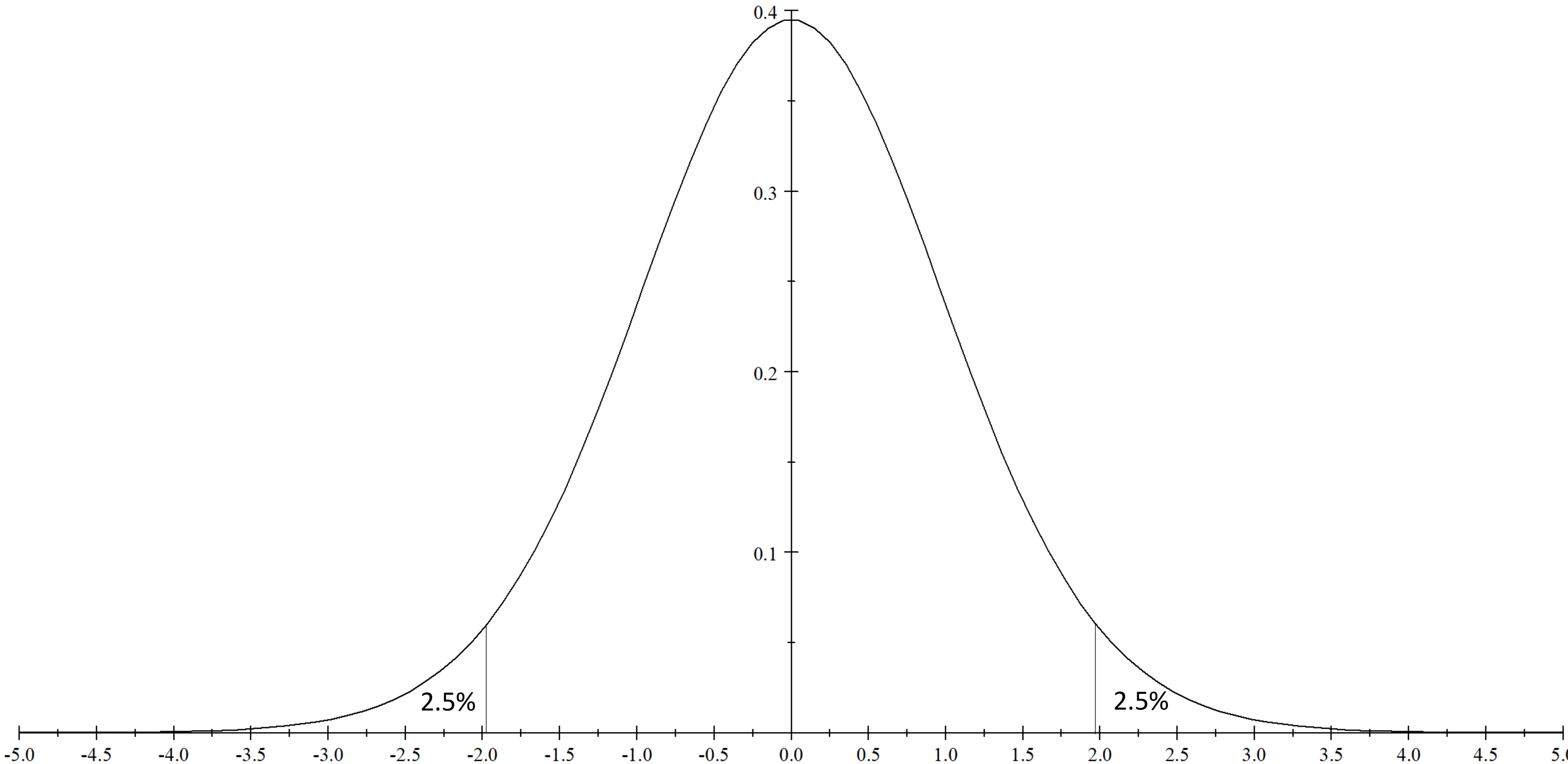
where $t_{\frac{\alpha}{2}}$ is the respective quantile of the t-distribution

Example: When $\alpha = 0.05$ we compare $\left|t_{\hat{\beta}}\right|$ with $t_{\frac{\alpha}{2}} = 1.962$ for $n - k - 1 = 1000$

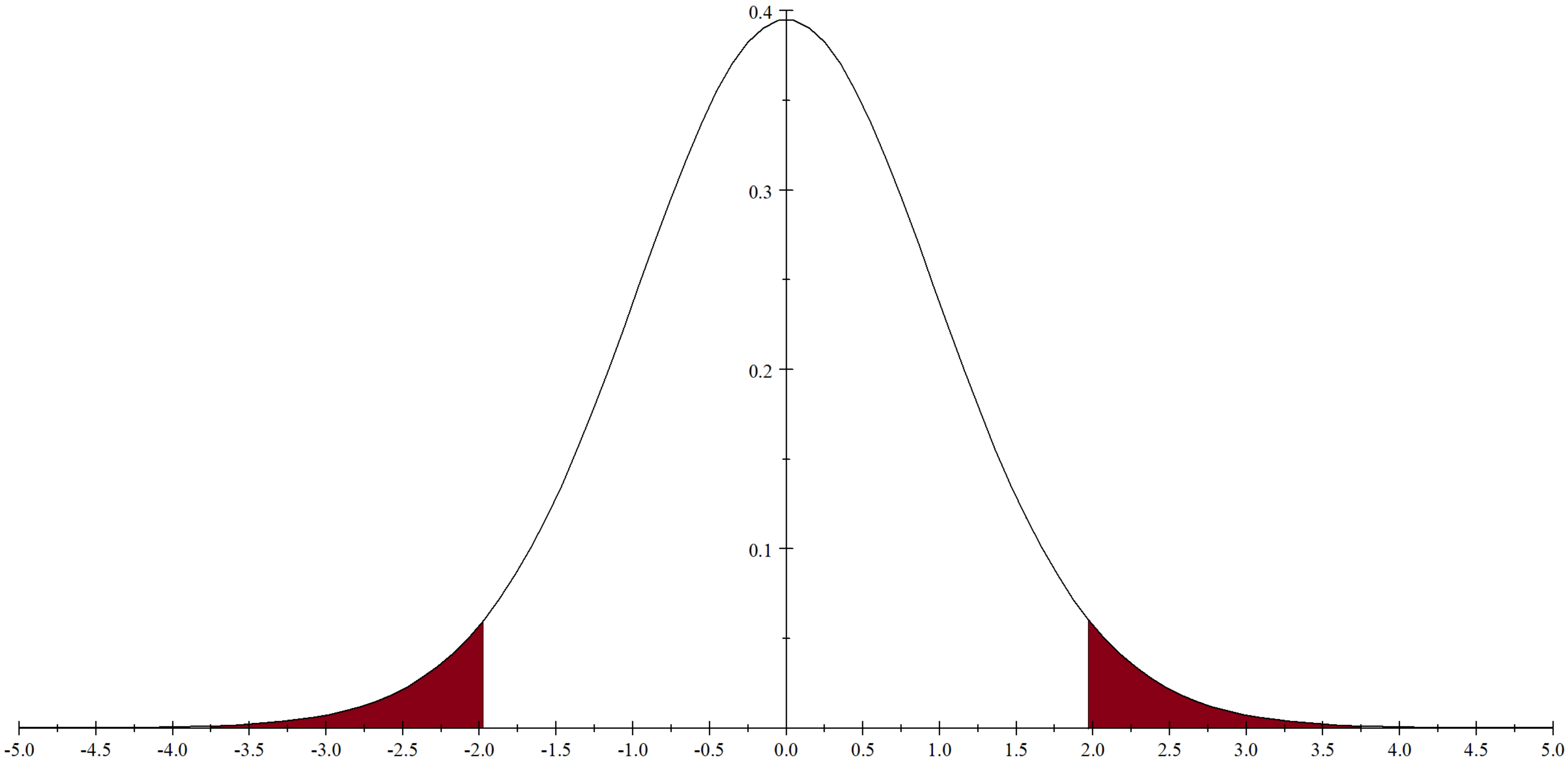
Consider the t-distribution under the Null hypothesis (if the true $\beta = 0$)



The likelihood that we will observe a t-value above 1.96 or below -1.96 is 5%:



We reject the Null hypothesis at a significance level of 5% if we observe a t-value above 1.96 or below -1.96



- Standard errors reported in regression tables yield the (estimated) standard deviation of the estimated $\hat{\beta}$
- The standard errors are used to construct the t-statistics
- From that we can compute the p-values (reported by statsmodels automatically)

Intuition: This gives us the following

- If I would draw different samples (of the given size) I would obtain different estimators $\hat{\beta}$
- What is the standard deviation of these $\hat{\beta}$?
- When this is small: we are close to the true β
- When this is large: there is much noise and therefore it is likely that the estimated $\hat{\beta}$ is further away from the true β

What a p-values tell us:

- What is the probability of obtaining an estimate that is at least as "extreme" (distant from 0) as the value of $\hat{\beta}_j$ I have estimated when the true value of were $\beta = 0$
- When this probability is smaller (computed using the estimated standard errors) we can be more certain that the true β is not zero
- For instance, when $p < 0.05$ for a certain coefficient, we say that the coefficient is *statistically significant at the* 5% *level*

- We mark this in a typical regression table in a paper with
  - * if $p < 0.10$
  - ** if $p < 0.05$ and
  - *** if $p < 0.01$

Your Task — Simulated data set

Create a new notebook in which you generate a data set with 100 observations where we know that the CEF is $y = 200 + 2x$:

- Create a variable which sets the number of observations:

```
n=100
```

- Create DataFrame with n rows and columns x and y:

```
df=pd.DataFrame(index=range(n), columns=['x','y'])
```

- Set x to a vector of n normally distributed random variables:

```
df['x']=np.random.normal(100,15,n)
```

- Set y according to the above CEF and add some noise:

```
df['y']=200+2*df['x'] + np.random.normal(0,500,n)
```

- Add a regression of `y` on `x`

Your Task

Simulated data set

- Run the script several times
(each time is like drawing a new sample from the population)
- Compare the regression estimates
  - Compare the estimated coefficients
  - Also look at the standard errors, t-values and p-values
- Increase the number of observations to 10.000 (`n=10000`)
- Repeat the exercise
- Save the notebook as `SimulateData`

Your Task **Simulated data set**

- Change your do file to generate „pure noise“

```
df['y']=200 + np.random.normal(0,500,n)
```

- Run the do file 20 times
(each time is like drawing a new sample from the population)

- Count the number of times you obtain a p-value for the coefficient of `x` that is smaller than *0.1*

3.2 (Robust) Standard Errors

- We made the assumption that
  - the residuals are normally distributed
  - they have the same variance for all observations
- In general we cannot be sure that this assumption that $e \sim N\left(0, \sigma^2 I_N\right)$ will hold
  - Implies that residuals are normally distributed and
  - that variance of residuals is constant → so-called *Homoscedasticity*
- But we can check to what extent this seems plausible
  - Graphically: plotting the residuals
  - Statistically: test the Null hypothesis that the residuals are homoscedastic (Breusch-Pagan test)

- But it always holds that

$$\hat{\beta} = \beta + \left[\sum_{i=1}^{N} X_i\, X_i'\right]^{-1} \sum_{i=1}^{N} X_i\, e_i$$

  That is $\hat{\beta}$ is the sum of the true β plus a function of the residual
- One can now show (see Angrist/Pischke, p. 45) that $\hat{\beta}$ is asymptotically normally distributed
  - with probability limit β (i.e. sample size grows → $\hat{\beta}$ comes closer to β)
  - and a covariance matrix that can be estimated from X_i and the residuals
- So called "Robust" standard errors follow from this covariance matrix
- Such robust standard errors are reported in StatsModels if you use

```
reg= smf.ols('y~X', data=df).fit(cov_type='HC1')
```

- Are called *robust* because
  - they are derived without assuming that the variance of the residuals is independent of X_i (i.e. they allow for heteroscedasticity)
  - and in large enough samples they provide accurate hypothesis testing without further distributional assumptions

Interdependent Observations & Clustered Standard Errors

- Standard errors are estimated under the assumption that
  - data are *independent observations*
  - or in other words each observation is a random draw from the population
- Very often this is not the case, for instance when
  - We observe several employees that come from the same firm
  - Or we observe the same employee at different dates
- The residuals will be correlated when the observations come from the same person or colleagues in the same firm
- Standard errors that are estimated assuming independence of employee observations are then biased
  - They are typically too small
  - p-Values are then smaller than they should be
  - We run into the danger of rejecting the null hypothesis too often

Two possible solutions:

- Use only firm level observations
  - For instance build a data set that includes one observation per firm
  - (in the above example we would have 1.000 observations)
  - And use the average job satisfaction in the firm as dependent variable
  - Hint: Pandas `groupby()` helps to build these aggregated data sets

- Use a different method for estimating standard errors
  - There are procedures that account for the interdependence of observations within groups or clusters
  - With StatsModels use
  - `reg= smf.ols('y~X', data=df).fit(cov_type='cluster', cov_kwds={'groups’: df['groupvar’]})`

Your Task — Clustered Standard Errors

- Open your ManagementPractices.py file
- Run the script
- Inspect the DataFrame
- Note:
  - For each firm (account_id) there are observations from different years
  - These observations will not be independent and thus standard errors will be biased
- Copy your regression commands to have them twice in the script
- In the second `smf.ols…` estimate cluster robust standard errorts adding

```
.fit(cov_type='cluster', cov_kwds={'groups': df['account_id']})
```

- Compare the standard errors in the two regressions

Appendix

Consider

$$\widehat{\beta_1} = \frac{\frac{1}{N}\sum_{i=1}^{N}(X_i - \bar{X})\,(Y_i - \bar{Y})}{\frac{1}{N}\sum_{i=1}^{N}(X_i - \bar{X})^2}$$

We can rewrite this as

$$\frac{\sum_{i=1}^{N}(X_i - \bar{X})\,(\beta_0 + \beta_1 X_i + e_i - (\beta_0 + \beta_1 \bar{X} + \bar{e}))}{\sum_{i=1}^{N}(X_i - \bar{X})^2}$$

$$= \frac{\sum_{i=1}^{N}(X_i - \bar{X})\,\beta_1 (X_i - \bar{X})}{\sum_{i=1}^{N}(X_i - \bar{X})^2} + \frac{\sum_{i=1}^{N}(X_i - \bar{X})\,e_i}{\sum_{i=1}^{N}(X_i - \bar{X})^2} - \frac{\sum_{i=1}^{N}(X_i - \bar{X})}{\sum_{i=1}^{N}(X_i - \bar{X})^2}\,\bar{e}$$

and as $\sum_{i=1}^{N}(X_i - \bar{X}) = 0$ we obtain

$$\hat{\beta} = \beta_1 + \frac{\sum_{i=1}^{N}(X_i - \bar{X})\,e_i}{\sum_{i=1}^{N}(X_i - \bar{X})^2}$$

→ The estimate is the sum of the population value and a function of the residuals

Take

$$\hat{\beta} = \beta_1 + \frac{\sum_{i=1}^{N}(X_i - \bar{X})\, e_i}{\sum_{i=1}^{N}(X_i - \bar{X})^2}$$

When the residuals are independently normally distributed and have the same variance $e \sim N\left(0, \sigma^2\right)$

$$V\,[\hat{\beta} \mid X_1, X_2, \dots, X_N] = \frac{1}{\left(\sum_{i=1}^{N}(X_i - \bar{X})^2\right)^2} V\left[\sum_{i=1}^{N}(X_i - \bar{X})\, e_i\right]$$

$$= \frac{\sum_{i=1}^{N}(X_i - \bar{X})^2 V[e_i]}{\left(\sum_{i=1}^{N}(X_i - \bar{X})^2\right)^2} \quad = \quad \frac{\sigma^2}{\sum_{i=1}^{N}(X_i - \bar{X})^2}$$

Hence

$$\hat{\beta} \sim N\left(\beta, \frac{\sigma^2}{\sum_{i=1}^{N}(X_i - \bar{X})^2}\right)$$