The Role of the Measure of Functional Complexity in Effort Estimation

Luigi Lavazza
Dipartimento di Informatica e Comunicazione
Università degli Studi dell'Insubria
Varese, Italy

Gabriela Robiolo
Facultad de Ingeniería
Universidad Austral
Buenos Aires, Argentina

Contents of the presentation

- Motivations
- Goal
- Method
- Case study
- Experimentation
- Conclusion

Motivations

- It is well known that effort does not depend exclusively on the functional size of the application
- Most widely used models are of the type:

  Effort=f(Size, <product characteristics parameters>, <process characteristics parameters>)

  where <product characteristics parameters> typically include some measure of **complexity**.

Goal

- The paper evaluates:
  - different types of functional size measures as effort estimators
  - the consequences of taking into consideration also the amount and complexity of required elaboration in the effort estimation

Method

- We take into consideration :
  - functional size measures (FP, CFP and UCP)
  - elaboration complexity measure (Paths)
- Evaluate how well these measures are correlated with the development effort.
- Measured a set of 17 projects and analyzed the resulting data

Method - Path

- Paths is a simple measure of complexity.
- Based on:
  - Information typically available from use case descriptions
  - on the application of the principles of McCabe's complexity measure to the descriptions of use cases in terms of scenarios.

Method – Path - Description of a use case

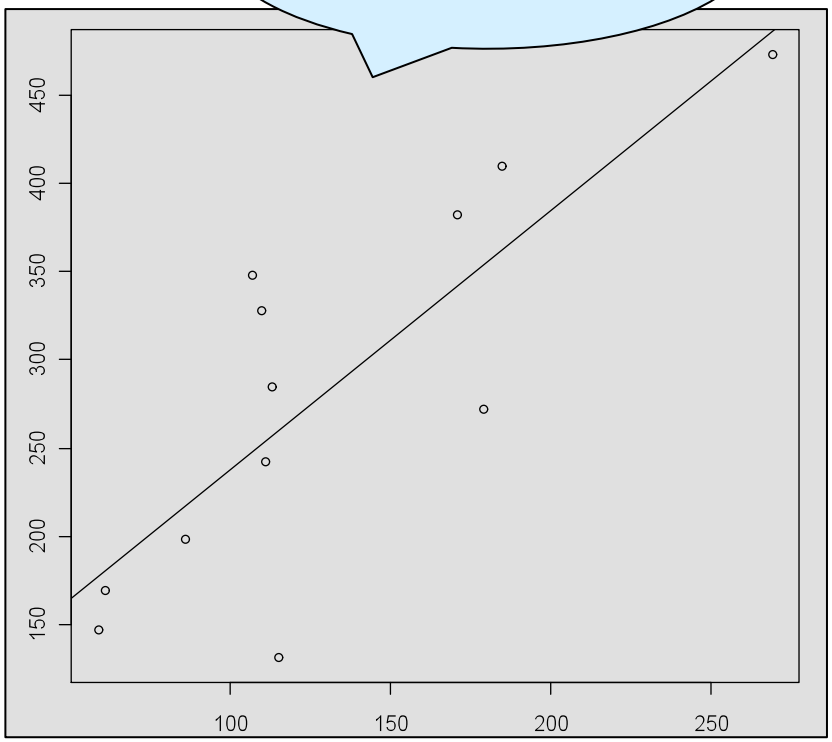
- The system displays a screen with the list of categories and subcategories.
- The user must choose the category to which the new product belongs. The list of properties of the selected categories and subcategories will appear below. ***If*** ***the*** categories and subcategories are new, a detailed description is also displayed.
- By pressing enter, the user will go to a new screen in which he should fill in gaps with the definition of the properties of the product.
- The use case finishes once the user enters a product, saves the changes, and leaves the system or cancels the operation.

Case study

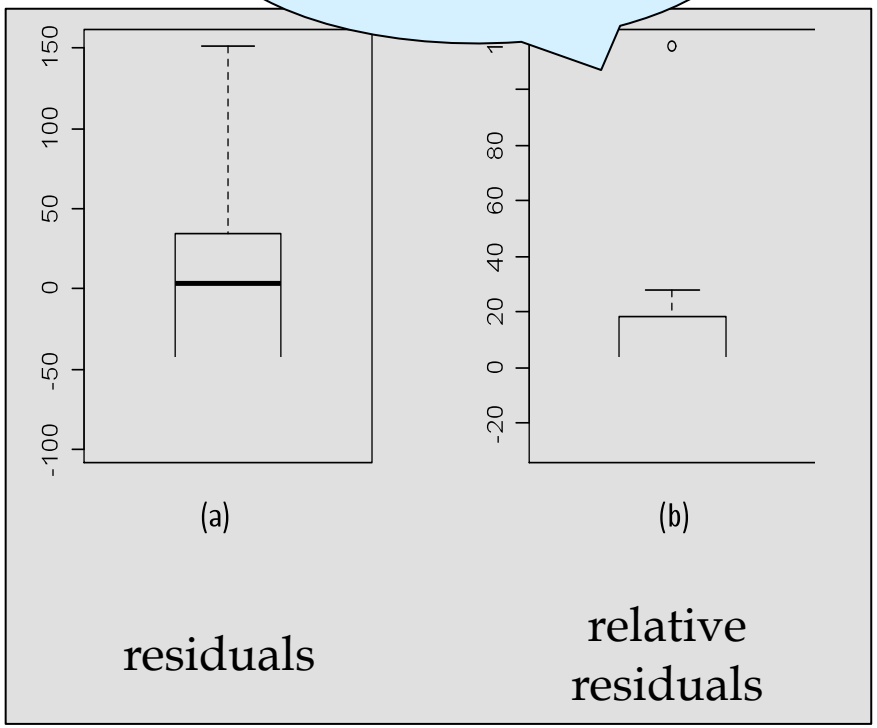
ID	Effort [PH]	Path	UFP	UCP	CFP
P1	410	71	185	201	129
P2	473.5	73	269	149	115
P3	382.4	60	171	84	108
P4	285	49	113	72	74
P5	328	34	110	72	48
P6	198	35	86	62	66
P7	442.02	50	75	71	81
P8	722.65	97	214	175	116
P9	392	83	340	111	119
P10	272	42	179	119	73
P11	131	18	115	68	51
P12	1042	118	168	169	85
P13	348	32	107	71	43
P14	242.5	68	111	99	113
P15	299.76	33	40	57	53
P16	147	20	59	53	53

Effort vs. UFP

R^2 = 0.62
p-value = 0.0014



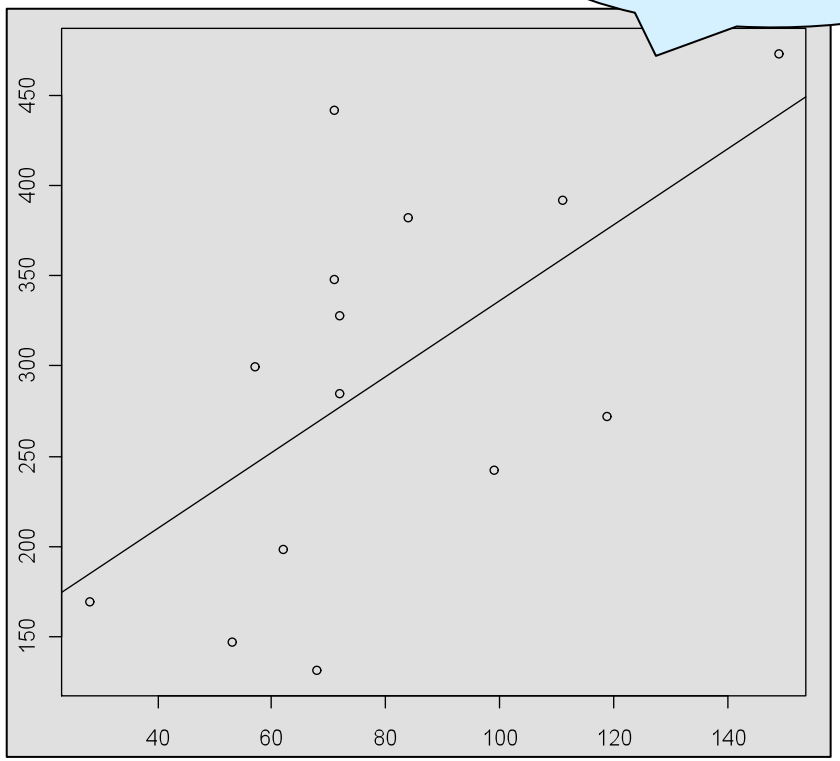
MMRE = 21.3%
Pred(25) = 75%



Effort = 1.4734 UFP + 90

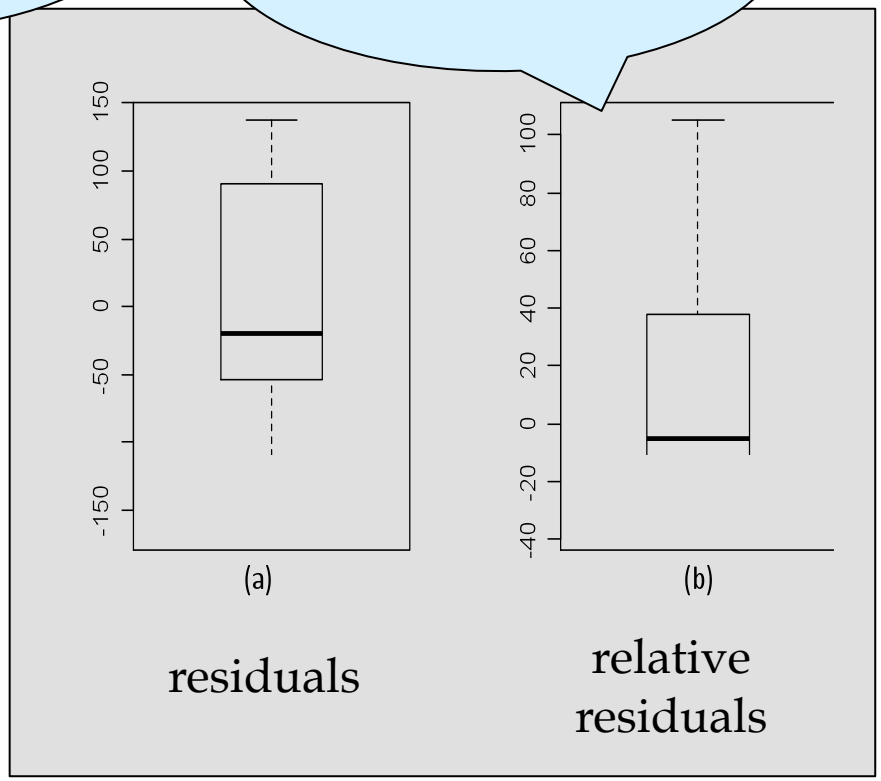
Effort vs. UCP

R^2 = 0.305
p-value = 0.024



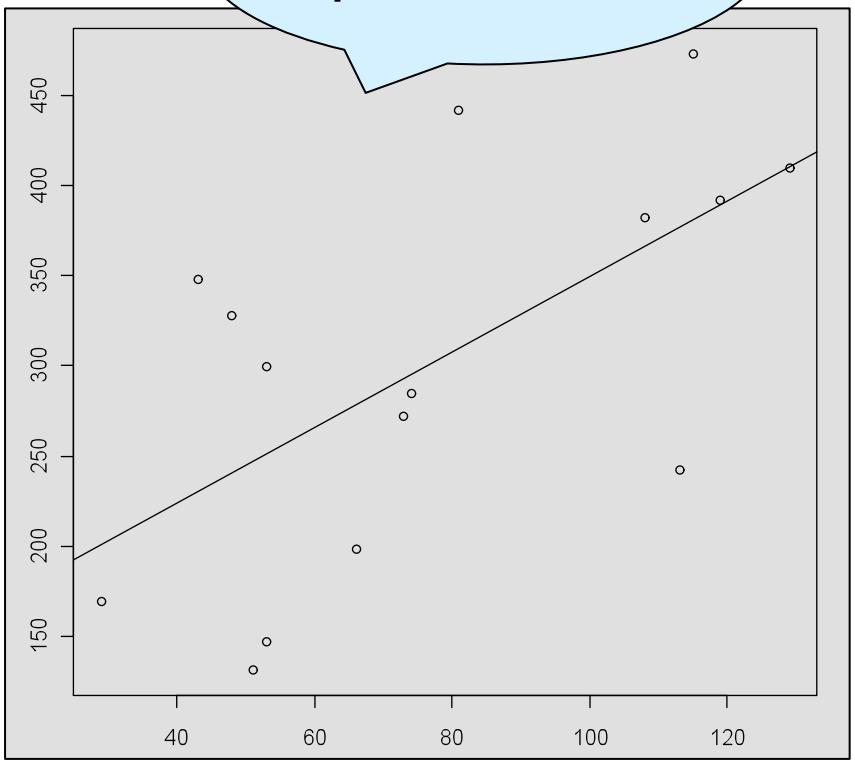
Effort = 2.1024 UCP + 126

MMRE = 29.5%
Pred(25) = 57.1%



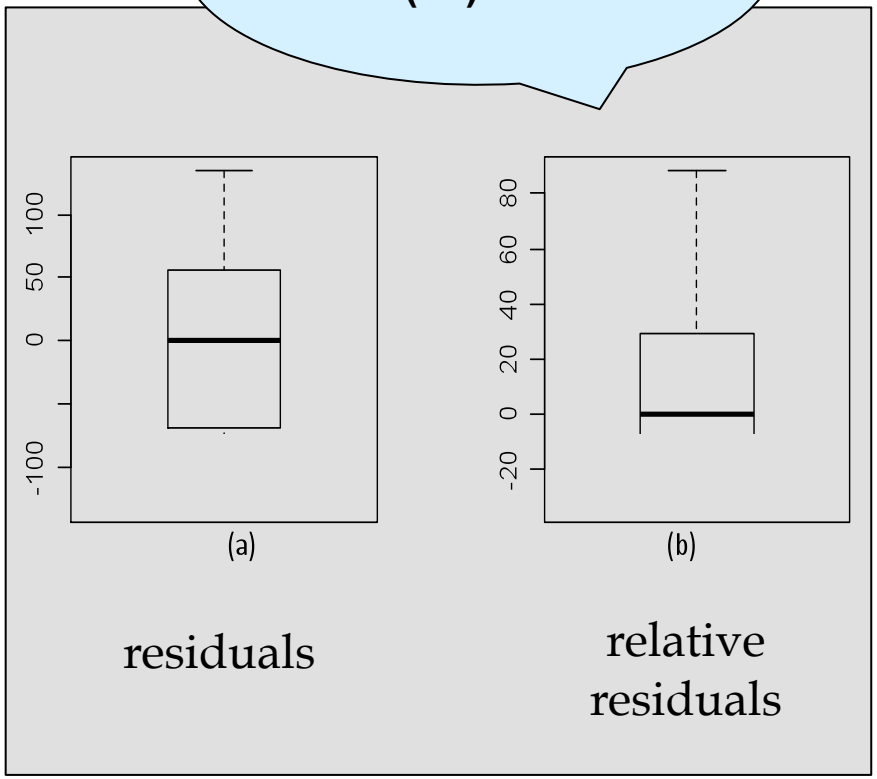
Effort vs. CFP

R^2= 0.34
p-value = 0.013



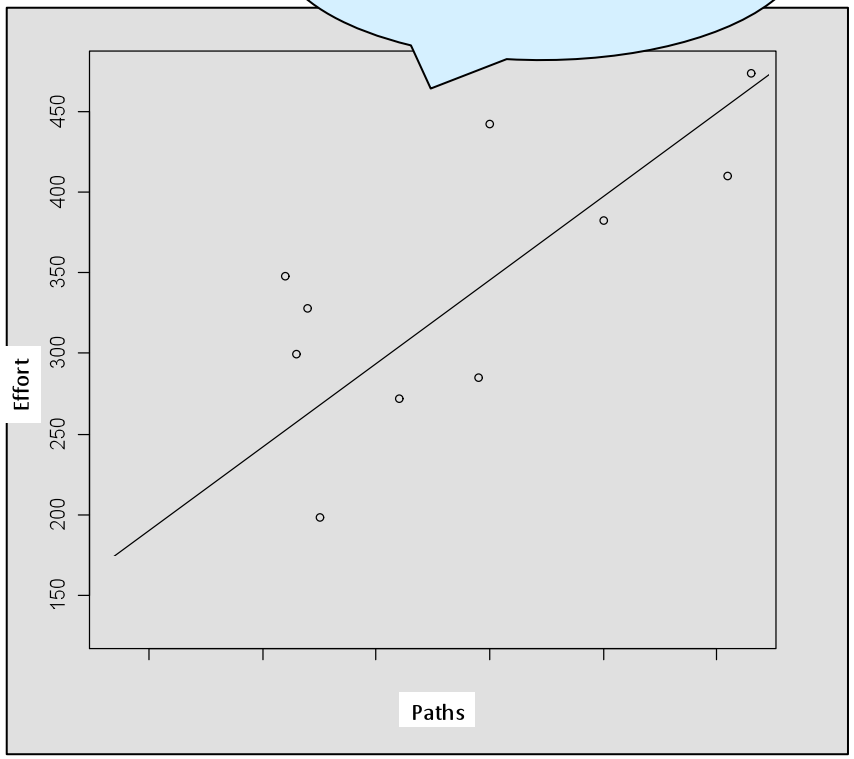
Effort = 2.104 CFP + 139

MMRE = 27.7%
Pred(25) = 53.3%

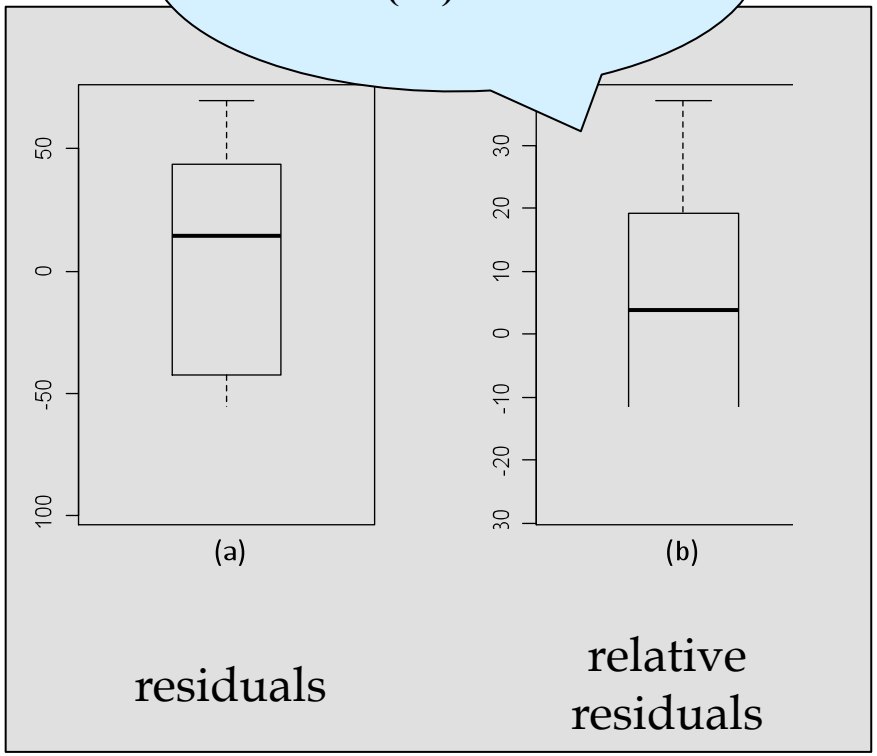


Effort vs. Paths

R^2= 0.714
p-value < 10^{-4}



MMRE = 18.1%
Pred(25) = 69.2%



Effort = 5.1769 Paths + 86

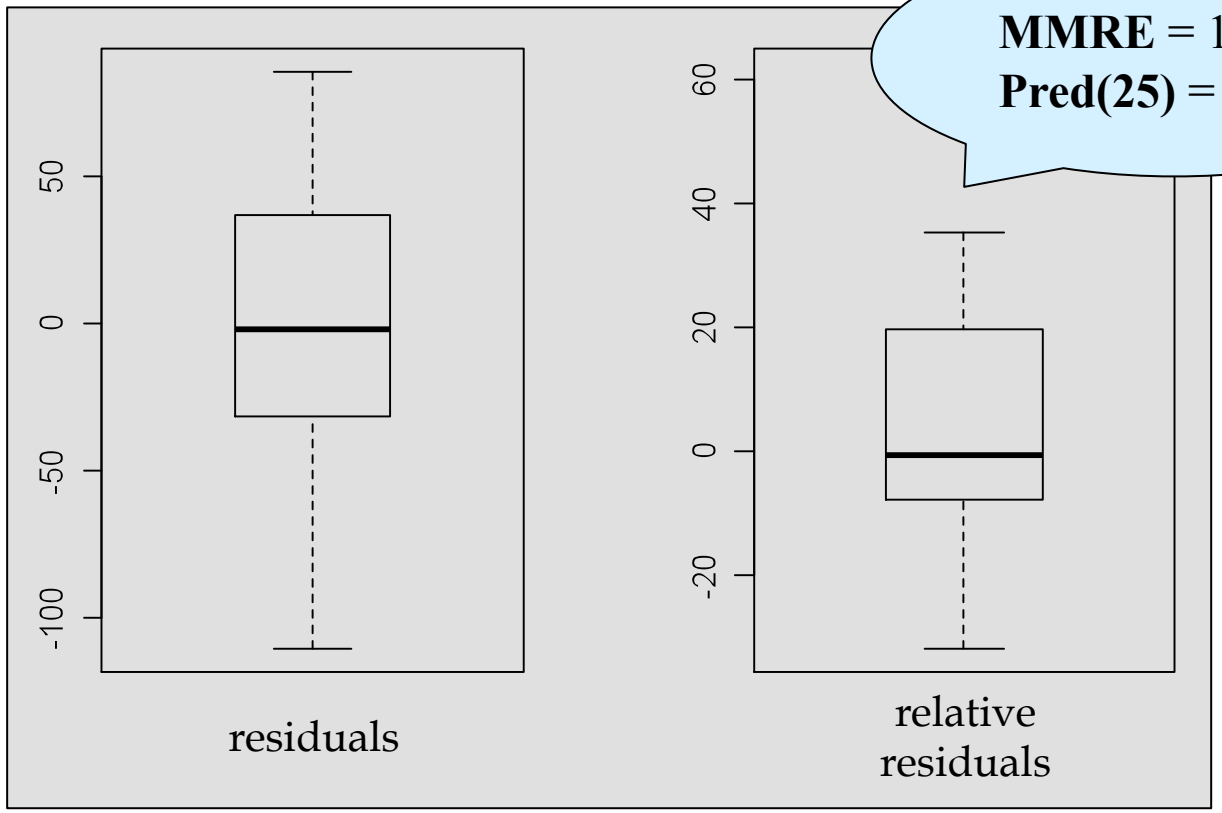
Effort vs. Functional Size and Complexity

- We do not use the number of Path
  - a big number of paths may indicate a complex application as well as a large one
  - Nonparametric tests indicate that Path is correlated with all the size measures (CFP, UFP, UCP)

- We use “Elaboration Density” as a measure of the complexity of the application or “amount of computation required”

$$\text{Elaboration Density} = \frac{\text{Number of Path}}{\text{Functional Size}}$$

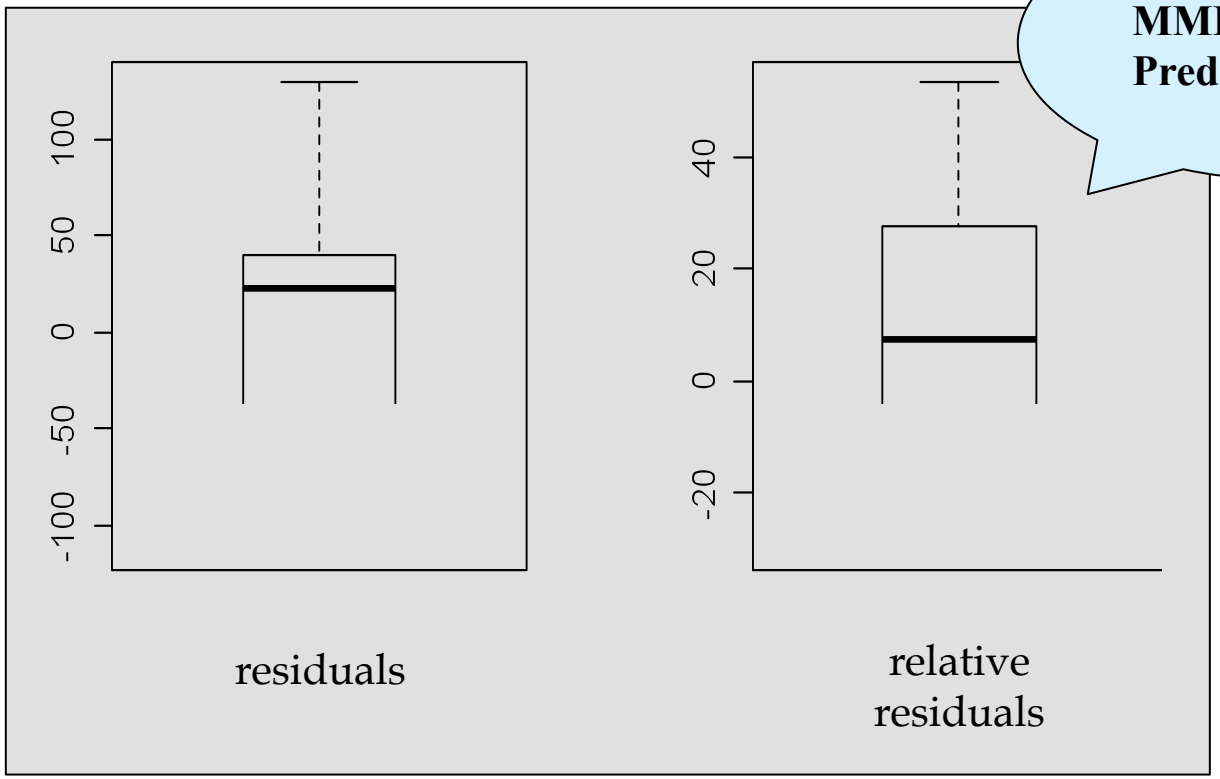
Effort vs. UFP and Complexity



R^2= 0.63 **p-value =** 0.003

Effort = 1.5117 UFP + 269.72 Paths/UFP - 5

Effort vs. UCP and Complexity

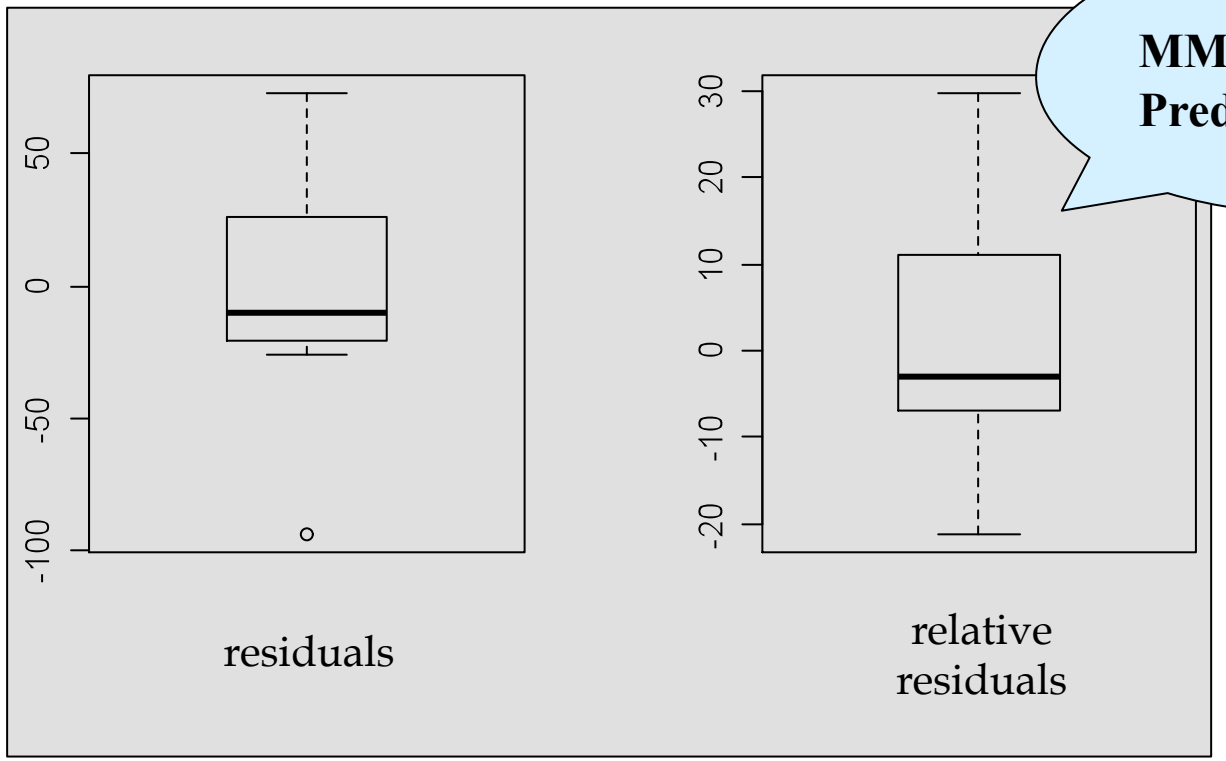


MMRE = 21.7%
Pred(25) = 53.6%

R^2= 0.5114 **p-value =** 0.005

Effort = 1.7802 UCP + 340.7816 Paths/UCP - 38

Effort vs. CFP and Complexity



MMRE = 12.1%
Pred(25) = 84.6%

R^2= 0.8278 **p-value <** 10^{-4}

Effort = 2.5967 CFP + 606.8266 Paths/CFP - 236

Effort vs. Functional Size and Complexity

Model	Adj. R^2	p-value	MMRE	Pred(25)	Error range
Effort vs. UFP	0.62	0.0014	21.3%	75%	-29%..98%
Effort vs. UCP	0.31	0.0235	29.5%	57.1%	-38%..105%
Effort vs. CFP	0.34	0.013	27.7%	53.3%	-34%..88%
Effort vs. Paths	0.71	$< 10^{-3}$	18.1%	69.2%	-28%..37%
Effort vs. UFP and Paths/UFP	0.63	0.0028	18.6%	69.2%	-32%.. 61%
Effort vs. UCP and Paths/UCP	0.51	0.0054	21.7%	53.6%	-30%..54%
Effort vs. CFP and Paths/CFP	0.83	$< 10^{-4}$	12.1%	84.6%	-21%..30%

Practical consequences

- The measures of functional size alone are not satisfactory predictors of the development effort
- Paths are better than size at predicting the development effort [Robiolo et al., ESEM 2009].
- Models of the type Effort=f(FS, Path/FS) are more precise than those using the functional size alone
- The model that uses CFP and Paths/CFP as independent variables outperforms the model that uses Paths alone as the independent variable

Threats to Validity

- The models involving UFP and UCP have some problems, from a statistical point of view:
  - The distribution of Paths/UFP is hardly normal, and the independent variables are mildly correlated (Spearman's r = -0.5);
  - The distribution of use case points is also hardly normal.
- The best model (i.e., Effort=f(CFP, Paths/CFP)) is perfectly valid from a statistical point of view.

Conclusions

- We considered the problem of measuring the amount of data elaboration required from an application according to its functional user requirements.
- The “elaboration density” expressed as Paths/CFP is adding to the model the idea of complexity per size unit.
- It appears useful to explain the amount of effort that is dedicated by developers to implementing data elaboration

Future Works

- Goals:
  - look for further evidence of Paths as a an effort predictor
  - comparing models based on CFP and Paths/CFP with other “state of the art” models, like COCOMO II or SEER/SEM
  - exploring the possibility of enhancing the definition of COSMIC FP by introducing a Paths-based measure of data elaboration in the notion of functional size.


UNIVERSITAS STUDIORUM INSUBRIAE

Backup slides