

RE-Pract 2018 Survey: Coding Instructions – Supplement

Table of Contents

VARIABLE 1373: RESEARCH WISHES.....	1
VARIABLE 18: FURTHER COMMENTS AND SUGGESTIONS	1
VARIABLE 6: PRIMARY WORKING AREA.....	2
VARIABLE 11: YEARS OF EXPERIENCE	3
VARIABLE 16: CLASS OF SYSTEM	4
VARIABLE 19: INDUSTRY SECTOR.....	5
VARIABLE 8345 ET SEQ: POSITIVE REASONING	6
VARIABLE 8780 ET SEQ: NEGATIVE REASONING	7
PAPER MAPPING: WHAT?.....	8
PAPER MAPPING: HOW?	9
PAPER MAPPING: WITH WHOM?	10

Variable 1373: Research Wishes

To Be Done

Variable 18: Further Comments and Suggestions

To Be Done

RE-Pract 2018 Survey: Coding Instructions – Supplement

Variable 6: Primary Working Area

Answers to Code: 29

Variable Type: List-Supplementing Short Text (for Variable 5)

Variable Question: Which of the following roles describes your primary working area best?
(Respondent answered “Other (please specify)” on Variable 5)

Options shown for Variable 5:

Summary of “Other (please specify)” after algorithmic coding:

- Requirements Engineer
- Business Analyst
- Architect
- Tester / Test Manager
- Project Manager
- Developer
- Product Owner
- Designer
- Other (please specify)

Ifdn	
v_6_coded	
Architect	3
Consultant	3
Context Roles	2
Designer	1
Manager	5
Multiple Roles	6
Process Designer	2
Researcher	7

Code used to produce this result:

```
def code_var_6(series):
    coded_series = []
    for value in series:
        value = value.lower()
        val = None
        if re.search('lecturer|phd\scandidate|researcher|r&d', value):
            val = 'Researcher'
        elif re.search('consultant', value):
            val = 'Consultant'
        elif re.search('systems?\sengineer', value):
            val = 'Architect'
        elif re.search('processes', value):
            val = 'Process Designer'
        elif re.search('design', value):
            val = 'Designer'
        elif re.search('marketing|iso\s\d+', value):
            val = 'Context Roles' # this was: 1 Marketing, 1 Regulator
        elif re.search('manag|cto', value):
            val = 'Manager'
        elif (re.search('different|changing|both|depend(?:s|ing)|combin',
            value) or (len(re.findall(',', value)) > 1)):
            val = 'Multiple Roles'
        else:
            raise Exception(f'Difficulty Coding Entry: {value}')
        coded_series.append(val)
    return coded_series
```

RE-Pract 2018 Survey: Coding Instructions – Supplement

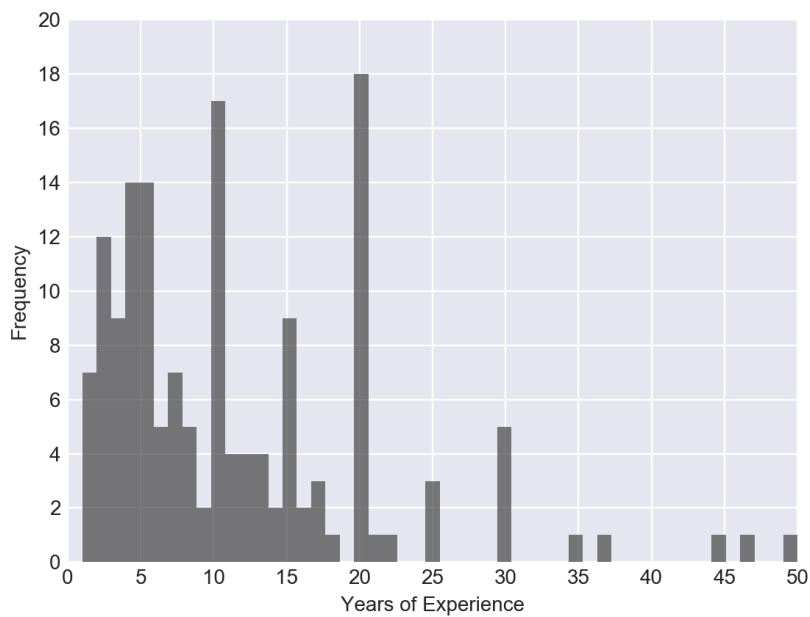
Variable 11: Years of Experience

Answers to Code: 154

Variable Type: List-Supplanting Short Text

Variable Question: How many years are you working in your primary working area?

Visual summary of responses after algorithmic coding:



Code used to produce this result:

```
def code_var_11(series):
    coded_series = []
    replace_dict = {'years?\\.??|y(?:\\w)|about': '',
                    'six': '6',
                    'one': '1',
                    ',': ',',
                    '\\+|>': ''}

    this_year = 2018
    for value in series:
        value = value.lower()
        val = None
        try:
            val = float(value)
        except:
            val = value
            for k, v in replace_dict.items():
                val = re.sub(k, v, val)
            try:
                val = float(val)
            except:
                if re.search('since\\s(\\d{4})', val):
                    val = 2018 - float(re.search('since\\s(\\d{4})', val).group(1))
                elif re.search('\\d+', val):
                    val = sum([float(x) for x in re.findall('\\d+', val)])
                else:
                    raise Exception(f'Difficulty Coding Entry: {value}')
            coded_series.append(val)
    return coded_series
```

RE-Pract 2018 Survey: Coding Instructions – Supplement

Variable 16: Class of System

Answers to Code: 12

Variable Type: List-Supplementing Short Text (for Variable 15)

Variable Question: What class of systems is in scope of your project(s)?
(Respondent answered “Other (please specify)” on Variable 15)

Options shown for Variable 15 were:

- Software-intensive embedded systems
- (Business) information systems
- Hybrid / mix of embedded systems and information systems
- Other (please specify)

Summary of “Other (please specify)” after algorithmic coding:

	lfdn
v_16_coded	
(Business) information systems	5
Hardware	3
Hybrid / mix of embedded systems and information systems	4

Code used to produce this result:

```
def code_var_16(series):
    coded_series = []
    for value in series:
        value = value.lower()
        val = None
        if re.search('all.*?above', value):
            val = 'Hybrid / mix of embedded systems and information systems'
        elif (re.search('c(?:ustomer|onsumer)|online|information', value) # infosys
              or re.search('(?!\\w)erp(?:\\w)', value)): # infosys, special (and doubtful ;-))
            val = '(Business) information systems'
        elif re.search('machine|infrastructure|processor', value):
            # or would you want to class these as hybrid?
            val = 'Hardware'
        elif re.search('aeronautics|railway', value): # guessing this one
            val = 'Hybrid / mix of embedded systems and information systems'
        else:
            raise Exception(f'Difficulty Coding Entry: {value}')
        coded_series.append(val)
    return coded_series
```

RE-Pract 2018 Survey: Coding Instructions – Supplement

Variable 19: Industry Sector

Answers to Code: 154

Variable Type: List-Supplanting Short Text

Variable Question: What is the industry sector in which you are most frequently involved?

Summary of responses after coding:

Code used to produce this result:

	lfdn
v_19_coded	
Academia	2
Aeronautics	7
Automation	3
Automotive	21
Consulting	2
E-Commerce	3
Education	7
Energy	4
Financial Services	20
Hardware	5
Healthcare	10
ICT	21
Infrastructure	6
Multiple Sectors	19
Public Sector	11
Software	5
Tourism	1
Transportation	7

```
def code_var_19(series):
    coded_series = []
    for value in series:
        value = value.lower()
        val = None
        if (re.search('mixed|varies(?:\w)|several|ecosystem|(?!\s)services(?:\s)',
                    value)
            or (len(re.findall(',', value)) > 1) or re.search('and', value)
            and not re.search('oil.*?gas|bank.*?fin|ins.*?bank|aero.*?defen|well.*heal', value)):
            val = 'Multiple Sectors'
        elif re.search('university|research|academi', value):
            val = 'Academia'
        elif re.search('aero|avi(?:on|at)', value):
            val = 'Aeronautics'
        elif re.search('automation', value):
            val = 'Automation'
        elif re.search('automotive', value):
            val = 'Automotive'
        elif re.search('consult', value):
            val = 'Consulting'
        elif re.search('e\-\?commerc|online', value):
            val = 'E-Commerce'
        elif re.search('educati', value):
            val = 'Education'
        elif re.search('energy|(?oil|gas)(?:\w)', value):
            val = 'Energy'
        elif re.search('financ|banki|insuran', value):
            val = 'Financial Services'
        elif re.search('semiconductor|robotics|computer\sengin|industrial\ssys', value):
            val = 'Hardware'
        elif re.search('medic(?:al|ine)|heal?th|wellness', value):
            val = 'Healthcare'
        elif re.search('railway|building|pipelines', value):
            val = 'Infrastructure'
        elif re.search('government|public\s(?:!transport)|defen[cs]e', value):
            val = 'Public Sector'
        elif re.search('software|saas', value):
            val = 'Software'
        elif re.search('transport|logis\w|marine', value):
            val = 'Transportation'
        elif re.search('tourism', value):
            val = 'Tourism'
        elif re.search(('commun|telecom|(?!\w)ict(?:\w)|(?!\w)it(?:\w)|(?!\w)iot(?:\w)|'
                        + 'intranet|electron|network|information'), value):
            val = 'ICT'
        else:
            raise Exception(f'Difficulty Coding Entry: {value}')
        coded_series.append(val)
    return coded_series
```

RE-Pract 2018 Survey: Coding Instructions – Supplement

Variable 8345 et seq: Positive Reasoning

While performing the validation, please watch for responses we might quote in our publications.

Summary of responses after algorithmic coding:

		Tag
level_1	level_2	
		31
NotAnswered		1
reason	originality	5
	plausibility	34
	relevance	63
source	experience	6
	opinion	12

Regular expressions used to produce this result:

```
posexes = {
  'reason:relevance':
    ('problem|challeng|experienc|issue(?:s)|concern|need(?:s)|dilemma|'+
    'relevan|essential|critical|crucial|importa|difficult|(?:^(\\W)we\\W|fundamental)'),
  'reason:plausibility':
    'could|might|help(s|ful)?(?:\\w)|improves(?:!m)|(?!\\sto\\s)better',
  'reason:originality':
    'literature|gap|interesting',
  'source:experience':
    '(?:^(\\W)my\\W.*?(?:experience|work))',
  'source:opinion':
    '(?:^(\\W)my\\Wopinion(?:!:)|believ|think|feel',
}
```

RE-Pract 2018 Survey: Coding Instructions – Supplement

Variable 8780 et seq: Negative Reasoning

While performing the validation, please watch for responses we might quote in our publications.

Summary of responses after algorithmic coding:

		Tag
level_1	level_2	
		36
NotAnswered		1
reason	notconvincing	26
	notefficient	5
	notimportant	18
	notinteresting	5
	notoriginal	3
	notrealistic	7
	respondentattitude	1
	toocomplicated	6
	toospecialized	7
	toosubjective	1
	toovague	3
rejection	questionnotunderstood	3
	ratingnotnegative	1

Regular expressions used to produce this result:

```
# nb some refer to problem some to solution some to question itself
negexes = {
  # not that 'not important' refers to absolute and relative (=prioritization) importance reasonings
  'reason:notimportant': ('not?\W.{,50}(?:impor|relev|need|necess|frequ)|'
    + 'distract|decr.*?relev|not?\W.{,20}(priori|essent)|not.*?big.*?prob'),
  'reason:notefficient': 'effort',
  'reason:notinteresting': '(:not?\W.{,10}|un)interest',
  # not that 'not convincing' statements are quite diffuse - critique of assumptions, critique of procedures, ...
  'reason:notconvincing':
    ("(:n't|not?)\W.{,10}?(?:conv|impr)|not?\W.{,30}?(help|use)|(:n't|not?)\W.{,10}sense|"
    + "pointless|worse|harm|waste|fail|simplistic|obscure|impractical|not?\W.*?good|"
    + "(:ca|do)n't|not?)\W.{,10}work|"
    + "(:not?\W|don't|can't|can\s?not){,30}(?:value|benefit)"),
  'reason:notoriginal': 'already',
  'reason:notrealistic': 'to.{,5}(?:theoretic|acad)|skill|scenario|real-w',
  'reason:toocomplicated': "(:not\W|n't).*?\Wund|to.{,5}technical", # ie not understood
  'reason:toospecialized': 'to.{,10}specif|particular|special.*?domain|limit|should.*?wider|narrow',
  'reason:toovague': "fluffy|don't know.*?use|not.*?understood",
  'reason:toosubjective': 'subjecti',
  'reason:respondentattitude': 'attitude',
  'rejection:ratingnotnegative': 'not?\W.*?lower',
  'rejection:questionnotunderstood': 'not?\W.*?underst.*?quest|sorry'
}
```

RE-Pract 2018 Survey: Coding Instructions – Supplement

Paper Mapping: What?

Summary of papers after algorithmic coding:

As the code used to produce this result is quite voluminous, please see the Jupyter Notebook HTML for details regarding the rules used and the rationales behind individual facets or tags.

level_1	level_2	level_3	level_4	PaperID
what	challenge	content	all	5
			completeness	13
			consistency	8
			feasibility	4
			traceability	43
			unambiguousness	18
			understandability	15
		context	regulation	23
			uncertainty	28
		failure		9
	people	collaboration		9
			communication	20
		skills		16
			subjectivity	9
	problem			14
		process	automation	47
			deciding	11
			formalization	4
			improving	36
			prioritization	16
			standardization	17
			visualization	12
				14
	documentation	artifacts		1
		businessmodels		4
		diagrams		11
		featuremodels		7
		goalmodels		42
		naturallanguage		1
		prototypes		1
		statemachines		11
		usecases		4
		userstories		

level_1	level_2	level_3	level_4	PaperID
what	domain	organization	agile	11
			distributed	3
			lean	2
			outsourced	1
		sector	automotive	4
			energy	2
			health	4
			it	5
		systemclass	media	6
			mobile	4
			nanotechnology	1
			public	4
			subsea	1
			supplier	3
			adaptive	6
			bi	1
			complex	1
			embedded	2
			safetycritical	5
	general	framework		2
			research	6
		architecture		6
			functional	12
		goals		6
			quality	17
		quality	all	9
			performance	2
			reliability	6
			safety	26
			security	3
			sustainability	2
	information	rules		4
		scenarios		6
		systembehavior		3
		phase	analysis	15
			elicitation	45
			evaluation	35
			management	69
			specification	27
		region	china	1
			finland	1

RE-Pract 2018 Survey: Coding Instructions – Supplement

Paper Mapping: How?

Please validate “How?” in conjunction with “With Whom?”. The validation file contains entries for both questions, and at least one row per paper.

Summary of papers after algorithmic coding:

			PaperID
level_1	level_2	level_3	
how	engineering	analysis	1
		method	162
		technology	52
	perspective	experience	38
		opinion	11
		philosophy	1
		review	14
	science	interrogation	43
		intervention	37
		observation	81

Code used to produce this result:

```
def assign_engineering(summary):
    level_1 = ':engineering'

    level_2 = [':analysis', ':technology', ':method']

    level_3 = {':analysis': ['^a set of metrics'],
                ':technology': ['^a tool', '^a solution', '^a model',
                                '^a taxonomy', '^an ontology', '^a (?:modell?ing |specification )language',
                                '^a template', '(?!\\w)a blueprint', '^a (formal )?framework'],
                ':method': ['^a method', '^a process', '^a.{,15}technique', 'training program']}

    return assign_tag(level_1, level_2, level_3, summary)

def assign_science(summary):
    level_1 = ':science'

    level_2 = [':observation', ':intervention', ':interrogation']

    level_3 = {':observation': ['(?:multi.)?case|field) study', '(?:data.|document.)driven study',
                                'industrial evaluation', '^an analysis'],
                ':intervention': ['experiment(?:s|\\s)', 'project-based study',
                                'workshop-based industrial study', 'action research'],
                ':interrogation': ['interview-based study|study based on.{,30}interviews',
                                'questionnaire', '(?!literature )(?:online.)?survey']}

    return assign_tag(level_1, level_2, level_3, summary)

def assign_perspective(summary):
    level_1 = ':perspective'

    level_2 = [':philosophy', ':opinion', ':experience', ':review']

    level_3 = {':philosophy': ['conceptual framework'],
                ':opinion': ['^a discussion', '\\svision', 'roadmap\\s'],
                ':experience': ['experience report'],
                ':review': ['literature (?:survey|study|review)', 'state of the art report']}

    return assign_tag(level_1, level_2, level_3, summary)
```

RE-Pract 2018 Survey: Coding Instructions – Supplement

Paper Mapping: With Whom?

Please validate “With Whom?” in conjunction with “How?”. The validation file contains entries for both questions, and at least one row per paper.

Summary of papers after algorithmic coding:

			PaperID
level_1	level_2	level_3	
withwhom	laypeople	others	1
		students	28
	professionals	academics	2
		practitioners	30

Code used to produce this result:

```
def assign_all_withwhom(summary):
    level_1 = [':laypeople', ':professionals']
    level_2 = {':laypeople': [':students', ':others'],
               ':professionals': [':academics', ':practitioners']}
    level_3 = {':students': ['with students', 'with practitioners and students'],
               ':others': ['with crowd.?workers'],
               ':academics': ['with academics', 'with researchers',
                              'with students and academics'],
               ':practitioners': ['with practitioners', 'with students and practitioners']}
    tags = []
    for l1 in level_1:
        for l2 in level_2[l1]:
            if any([re.search(x, summary.lower()) for x in level_3[l2]]):
                tags.append('withwhom'+l1+l2)
    return tags
```