Data Quality

Alfonso R. Reyes

April 2019

Importance of Data Quality

Importance of Data Quality

Impact of not checking well input

- Misleading results
- Poor input, poor output
- Models will not converge
- Work has to be redone
- Will be unable not close the cycle
  - *cannot improve what we don't know*
- Lack of trust
  - *models get known to be unreliable*

Finding and filling missing data

Example of filling missing data

Well Tests

Well test data: numeric variables

```
#>  [1] "INSTRUMENTATION_TYPE"          "CALC_SEQ_NO"
#>  [3] "CALC_DC_SEQ_NO"                "DURATION_HRS"
#>  [5] "CHOKE_SIZE"                    "Z_SEP_PRESS_PSIG"
#>  [7] "Z_SEP_PRESS_BARG"              "Z_SEP_TEMP_C"
#>  [9] "Z_SEP_TEMP_F"                  "WH_PRESS_PSIG"
#> [11] "WH_PRESS_BARG"                 "ANNULUS_PRESS_PSIG"
#> [13] "ANNULUS_PRESS_BARG"            "WH_USC_PRESS_PSIG"
#> [15] "WH_USC_PRESS_BARG"             "WH_USC_TEMP_C"
#> [17] "WH_USC_TEMP_F"                 "WH_DSC_PRESS_PSIG"
#> [19] "WH_DSC_PRESS_BARG"             "GL_CHOKE_SIZE"
#> [21] "GL_RATE_SCFPERDAY"             "GL_RATE_SM3PERDAY"
#> [23] "GL_RATE_MSCFPERDAY"            "Z_TOTAL_GAS_SCFPERDAY"
#> [25] "Z_TOTAL_GAS_SM3PERDAY"         "Z_TOTAL_GAS_MSCFPERDAY"
#> [27] "TOT_WATER_RATE_ADJ_BBLSPERDAY" "TOT_WATER_RATE_ADJ_M3PERDAY"
#> [29] "NET_COND_RATE_ADJ_BBLSPERDAY"  "NET_COND_RATE_ADJ_SM3PERDAY"
#> [31] "GAS_RATE_ADJ_SCFPERDAY"        "GAS_RATE_ADJ_SM3PERDAY"
#> [33] "GAS_RATE_ADJ_MSCFPERDAY"       "NET_OIL_RATE_ADJ_BBLSPERDAY"
#> [35] "NET_OIL_RATE_ADJ_SM3PERDAY"    "LIQUID_RATE_ADJ_BBLSPERDAY"
#> [37] "LIQUID_RATE_ADJ_SM3PERDAY"     "GOR_SCFPERBBL"
#> [39] "Z_FGOR_SCFPERBBL"              "WOR"
#> [41] "Z_IGLR_SCFPERBBL"              "Z_TGLR_SCFPERBBL"
#> [43] "Z_GUF_SCFPERBBL"               "WATERCUT_PCT"
#> [45] "DRY_WET_GAS_RATIO"             "WGR_BBLPERSCF"
#> [47] "CGR_BBLPERSCF"                 "RESULT_NO"

#>  [1] "PRODUCTION_DAY"       "CODE"                 "OBJECT_START_DATE"
#>  [4] "OBJECT_END_DATE"      "WELL_CLASS"           "WELL_TYPE"
#>  [7] "FLOW_METHOD"          "PUMP_TYPE"            "WELL_METER_FREQ"
#> [10] "GAUGE_FLAG"           "COMP_AS_SEL_FLAG"     "CHOKE_UOM"
```

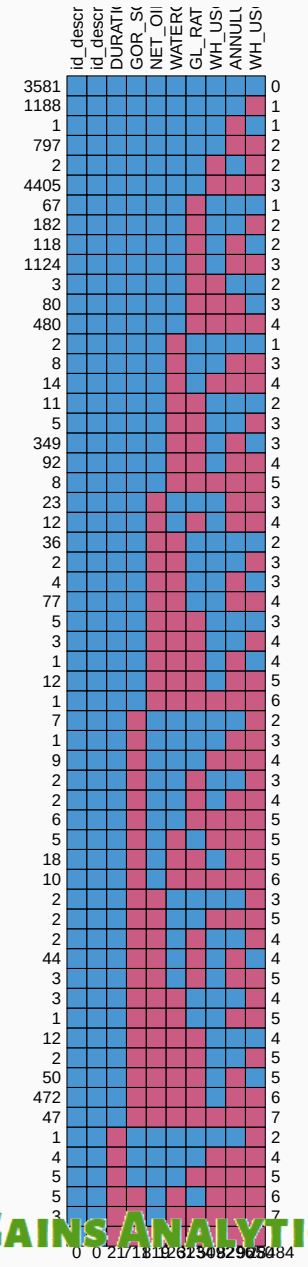
Look for missing values

In this case we are using the statistical package **mice** to fill in for the missing values.

```
library(tidyverse)
library(mice)
library(lattice)
```

```
# plot the missing data in a matrix by variables
md_pattern <- md.pattern(df, rotate.names = TRUE)
```

Look for missing values



Missing values per variable

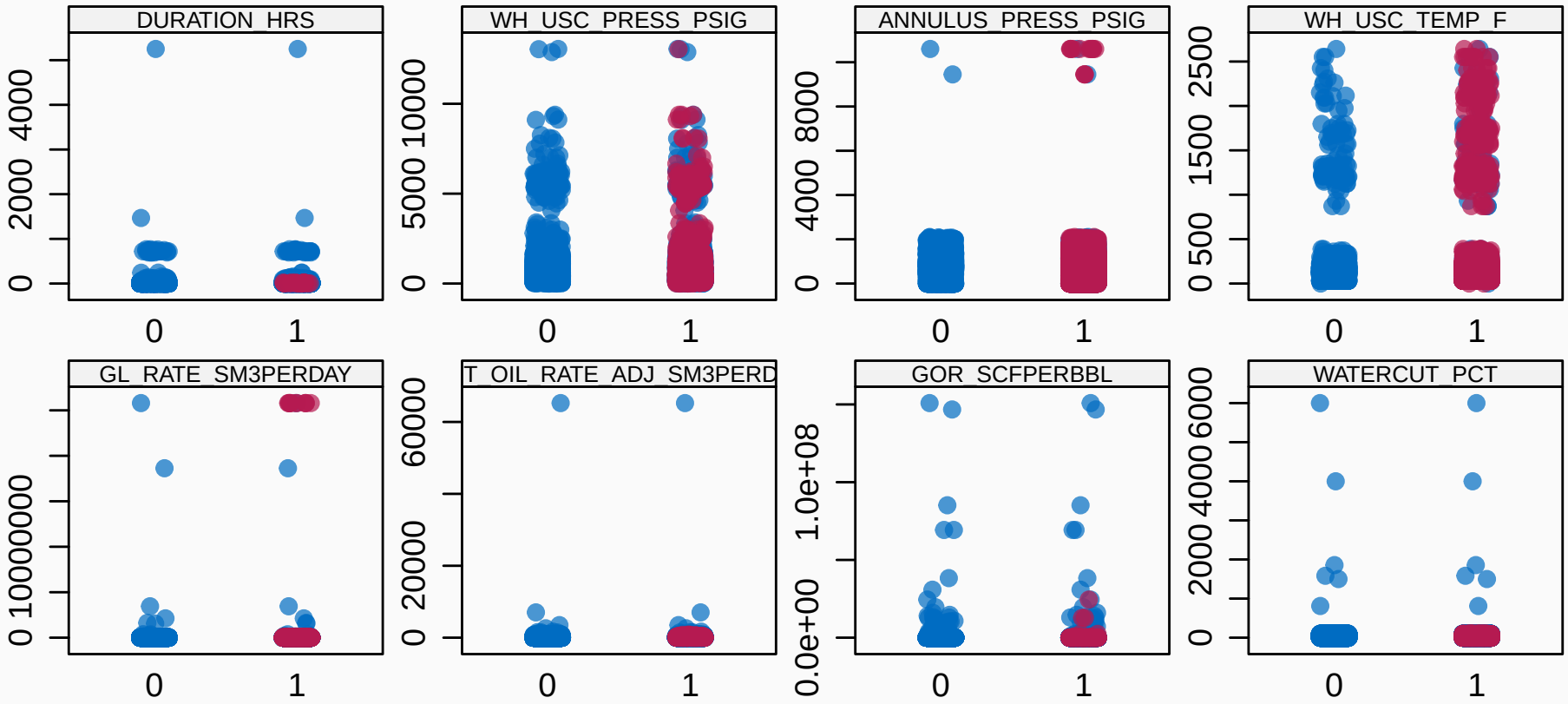
```
# number of complete cases: no variable value is missing in that observation
sum(complete.cases(sel_numdf))
```

```
#> [1] 3581
```

```
sel_numdf %>%
    skim_to_list() %>%
    .[["numeric"]] %>%
    as.data.frame(sapply(as.numeric)) %>%
    dplyr::select(variable, missing, complete)
```

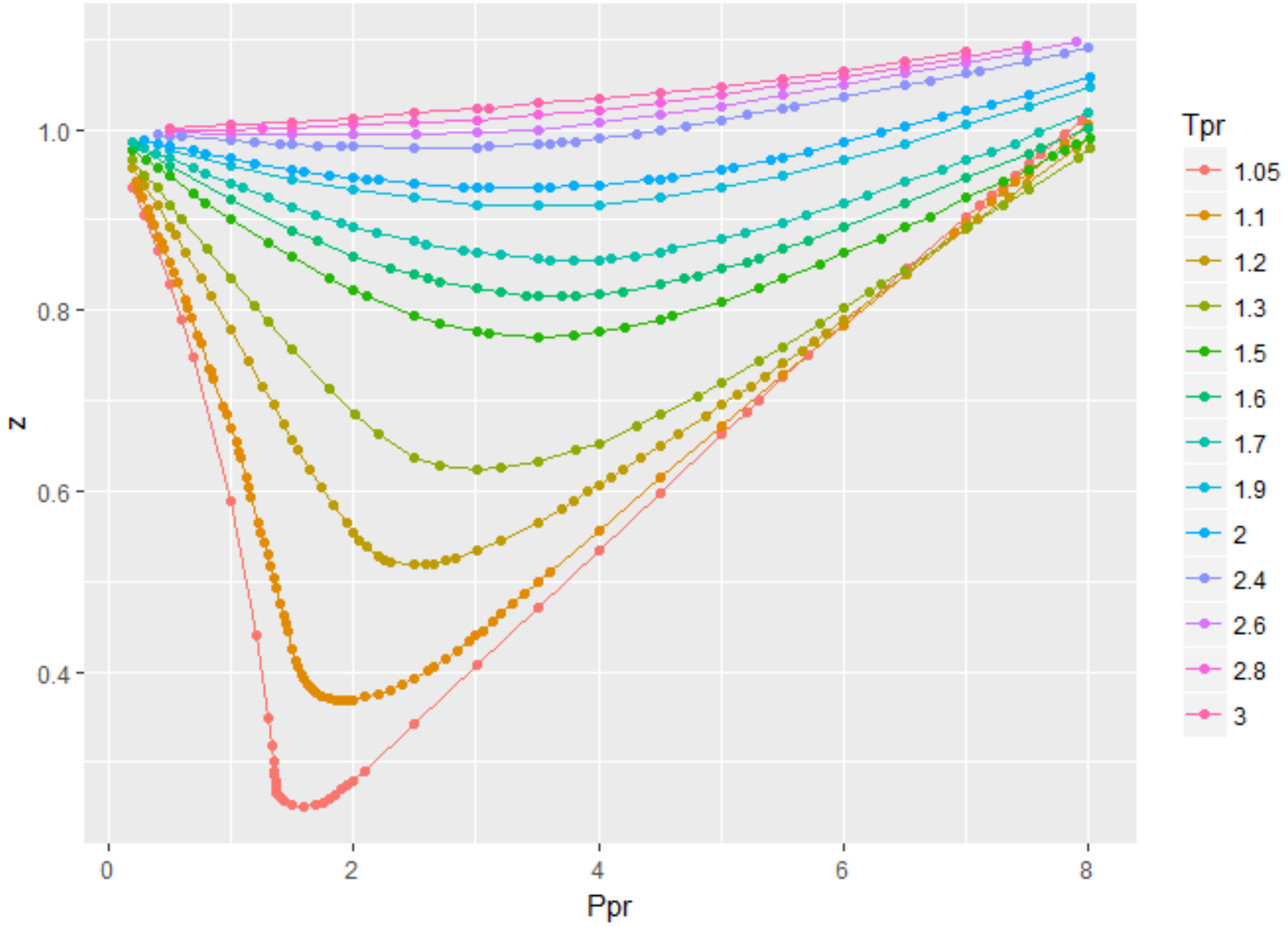
```
#>                    variable missing complete
#> 1         ANNULUS_PRESS_PSIG    8296     5118
#> 2              DURATION_HRS      21    13393
#> 3          GL_RATE_SM3PERDAY    3234    10180
#> 4              GOR_SCFPERBBL     711    12703
#> 5 NET_OIL_RATE_ADJ_SM3PERDAY     819    12595
#> 6              WATERCUT_PCT    1261    12153
#> 7          WH_USC_PRESS_PSIG    5092     8322
#> 8              WH_USC_TEMP_F    9050     4364
```

How well the fill data fits in



Tidy data

Standing-Katz chart



What is not tidy data

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Ppr	Tpr	z	Ppr	Tpr	z	Ppr	Tpr	z	Ppr	Tpr	z	Ppr	Tpr	z	Ppr	Tpr
2	0	1.05	1	0	1.1	1	0	1.15	1	0	1.2	1	0	1.25	1	0	
3	0.004023	1.05	1.00046	0.004023	1.1	1.00046	0.004023	1.15	1.00046	0.004023	1.2	1.00046	0.004023	1.25	1.00046	0.004023	
4	0.090309	1.05	0.973296	0.090309	1.1	0.975995	0.090309	1.15	0.978835	0.090309	1.2	0.982543	0.090309	1.25	0.984702	0.090309	
5	0.116359	1.05	0.965095	0.116359	1.1	0.968608	0.116359	1.15	0.972306	0.116359	1.2	0.977134	0.116359	1.25	0.979945	0.116359	
6	0.150927	1.05	0.953882	0.150927	1.1	0.958807	0.150927	1.15	0.963643	0.150927	1.2	0.969956	0.150927	1.25	0.973632	0.150927	
7	0.159339	1.05	0.951153	0.159339	1.1	0.956422	0.159339	1.15	0.961535	0.159339	1.2	0.968209	0.159339	1.25	0.972096	0.159339	
8	0.168049	1.05	0.948327	0.168049	1.1	0.953953	0.168049	1.15	0.959352	0.168049	1.2	0.966401	0.168049	1.25	0.970505	0.168049	
9	0.185494	1.05	0.942668	0.185494	1.1	0.949006	0.185494	1.15	0.95498	0.185494	1.2	0.962778	0.185494	1.25	0.96732	0.185494	
10	0.220061	1.05	0.931455	0.220061	1.1	0.939205	0.220061	1.15	0.946317	0.220061	1.2	0.955601	0.220061	1.25	0.961007	0.220061	
11	0.236999	1.05	0.925819	0.236999	1.1	0.934403	0.236999	1.15	0.942072	0.236999	1.2	0.952083	0.236999	1.25	0.957914	0.236999	
12	0.245987	1.05	0.922829	0.245987	1.1	0.931854	0.245987	1.15	0.939819	0.245987	1.2	0.950217	0.245987	1.25	0.956272	0.245987	
13	0.280276	1.05	0.91256	0.280276	1.1	0.922132	0.280276	1.15	0.931226	0.280276	1.2	0.943097	0.280276	1.25	0.95001	0.280276	
14	0.280549	1.05	0.912478	0.280549	1.1	0.922055	0.280549	1.15	0.931157	0.280549	1.2	0.94304	0.280549	1.25	0.949961	0.280549	
15	0.306029	1.05	0.904001	0.306029	1.1	0.91483	0.306029	1.15	0.924771	0.306029	1.2	0.937749	0.306029	1.25	0.945307	0.306029	
16	0.306059	1.05	0.903991	0.306059	1.1	0.914822	0.306059	1.15	0.924764	0.306059	1.2	0.937743	0.306059	1.25	0.945302	0.306059	
17	0.306476	1.05	0.903853	0.306476	1.1	0.914703	0.306476	1.15	0.92466	0.306476	1.2	0.937657	0.306476	1.25	0.945226	0.306476	
18	0.332189	1.05	0.895298	0.332189	1.1	0.907413	0.332189	1.15	0.918215	0.332189	1.2	0.932317	0.332189	1.25	0.94053	0.332189	
19	0.332402	1.05	0.895227	0.332402	1.1	0.907352	0.332402	1.15	0.918162	0.332402	1.2	0.932269	0.332402	1.25	0.940491	0.332402	
20	0.349698	1.05	0.887464	0.349698	1.1	0.902448	0.349698	1.15	0.913827	0.349698	1.2	0.928383	0.349698	1.25	0.937333	0.349698	
21	0.366736	1.05	0.881796	0.366736	1.1	0.897617	0.366736	1.15	0.909557	0.366736	1.2	0.924554	0.366736	1.25	0.934221	0.366736	
22	0.375094	1.05	0.879015	0.375094	1.1	0.895248	0.375094	1.15	0.907463	0.375094	1.2	0.922676	0.375094	1.25	0.932695	0.375094	
23	0.375625	1.05	0.878838	0.375625	1.1	0.895097	0.375625	1.15	0.90733	0.375625	1.2	0.922557	0.375625	1.25	0.932598	0.375625	
24	0.392509	1.05	0.87042	0.392509	1.1	0.89031	0.392509	1.15	0.903098	0.392509	1.2	0.918763	0.392509	1.25	0.929514	0.392509	
25	0.392926	1.05	0.870213	0.392926	1.1	0.890192	0.392926	1.15	0.902994	0.392926	1.2	0.918669	0.392926	1.25	0.929438	0.392926	
26	0.400946	1.05	0.867811	0.400946	1.1	0.887918	0.400946	1.15	0.900984	0.400946	1.2	0.916867	0.400946	1.25	0.927974	0.400946	
27	0.401284	1.05	0.86771	0.401284	1.1	0.887822	0.401284	1.15	0.900899	0.401284	1.2	0.916791	0.401284	1.25	0.927912	0.401284	
28	0.418847	1.05	0.86245	0.418847	1.1	0.882842	0.418847	1.15	0.896497	0.418847	1.2	0.912845	0.418847	1.25	0.924705	0.418847	
29	0.435503	1.05	0.856908	0.435503	1.1	0.878119	0.435503	1.15	0.892323	0.435503	1.2	0.909102	0.435503	1.25	0.921663	0.435503	
30	0.435831	1.05	0.856799	0.435831	1.1	0.878026	0.435831	1.15	0.892241	0.435831	1.2	0.909028	0.435831	1.25	0.921603	0.435831	
31	0.436009	1.05	0.85674	0.436009	1.1	0.877976	0.436009	1.15	0.892196	0.436009	1.2	0.908993	0.436009	1.25	0.92157	0.436009	
32	0.444566	1.05	0.853893	0.444566	1.1	0.875129	0.444566	1.15	0.890052	0.444566	1.2	0.907283	0.444566	1.25	0.920008	0.444566	
33	0.444774	1.05	0.853824	0.444774	1.1	0.87506	0.444774	1.15	0.890005	0.444774	1.2	0.907242	0.444774	1.25	0.91997	0.444774	
34	0.461737	1.05	0.845367	0.461737	1.1	0.869416	0.461737	1.15	0.886193	0.461737	1.2	0.903853	0.461737	1.25	0.916872	0.461737	
35	0.461936	1.05	0.845268	0.461936	1.1	0.86935	0.461936	1.15	0.886148	0.461936	1.2	0.903808	0.461936	1.25	0.916836	0.461936	
36	0.462075	1.05	0.845198	0.462075	1.1	0.869309	0.462075	1.15	0.886117	0.462075	1.2	0.903777	0.462075	1.25	0.91681	0.462075	
37	0.47008	1.05	0.842801	0.47008	1.1	0.866911	0.47008	1.15	0.884318	0.47008	1.2	0.901978	0.47008	1.25	0.915348	0.47008	
38	0.478602	1.05	0.840249	0.478602	1.1	0.864359	0.478602	1.15	0.882403	0.478602	1.2	0.900063	0.478602	1.25	0.913792	0.478602	
39	0.478835	1.05	0.840179	0.478835	1.1	0.864289	0.478835	1.15	0.882351	0.478835	1.2	0.900011	0.478835	1.25	0.91375	0.478835	
40	0.479113	1.05	0.840096	0.479113	1.1	0.864206	0.479113	1.15	0.882289	0.479113	1.2	0.899948	0.479113	1.25	0.913699	0.479113	

Standing-Katz | Katz (High P)

After tidying it up

	A	B	C	D
1	Tpr	Ppr	z	set
2	1.05	0	1	0
3	1.05	0.004023	1.00046	0
4	1.05	0.090309	0.973296	0
5	1.05	0.116359	0.965095	0
6	1.05	0.150927	0.953882	0
7	1.05	0.159339	0.951153	0
8	1.05	0.168049	0.948327	0
9	1.05	0.185494	0.942668	0
10	1.05	0.220061	0.931455	0
11	1.05	0.236999	0.925819	0
12	1.05	0.245987	0.922829	0
13	1.05	0.280276	0.91256	0
14	1.05	0.280549	0.912478	0
15	1.05	0.306029	0.904001	0
16	1.05	0.306059	0.903991	0
17	1.05	0.306476	0.903853	0
18	1.05	0.332189	0.895298	0
19	1.05	0.332402	0.895227	0
20	1.05	0.349698	0.887464	0
21	1.05	0.366736	0.881796	0
22	1.05	0.375094	0.879015	0
23	1.05	0.375625	0.878838	0
24	1.05	0.392509	0.87042	0
25	1.05	0.392926	0.870213	0
26	1.05	0.400946	0.867811	0
27	1.05	0.401284	0.86771	0
28	1.05	0.418847	0.86245	0
29	1.05	0.435503	0.856908	0
30	1.05	0.435831	0.856799	0
31	1.05	0.436009	0.85674	0
32	1.05	0.444566	0.853893	0
33	1.05	0.444774	0.853824	0
34	1.05	0.461737	0.845367	0

Why is so hard to operate with untidy data

- Difficult to vectorize
- Extra steps to perform arithmetic operations
- Difficult to plot: series, facets
- Plotting doesn't scale very well

Maybe comfortable for the human eyes ...

... but less effective for computers.

Detecting outliers

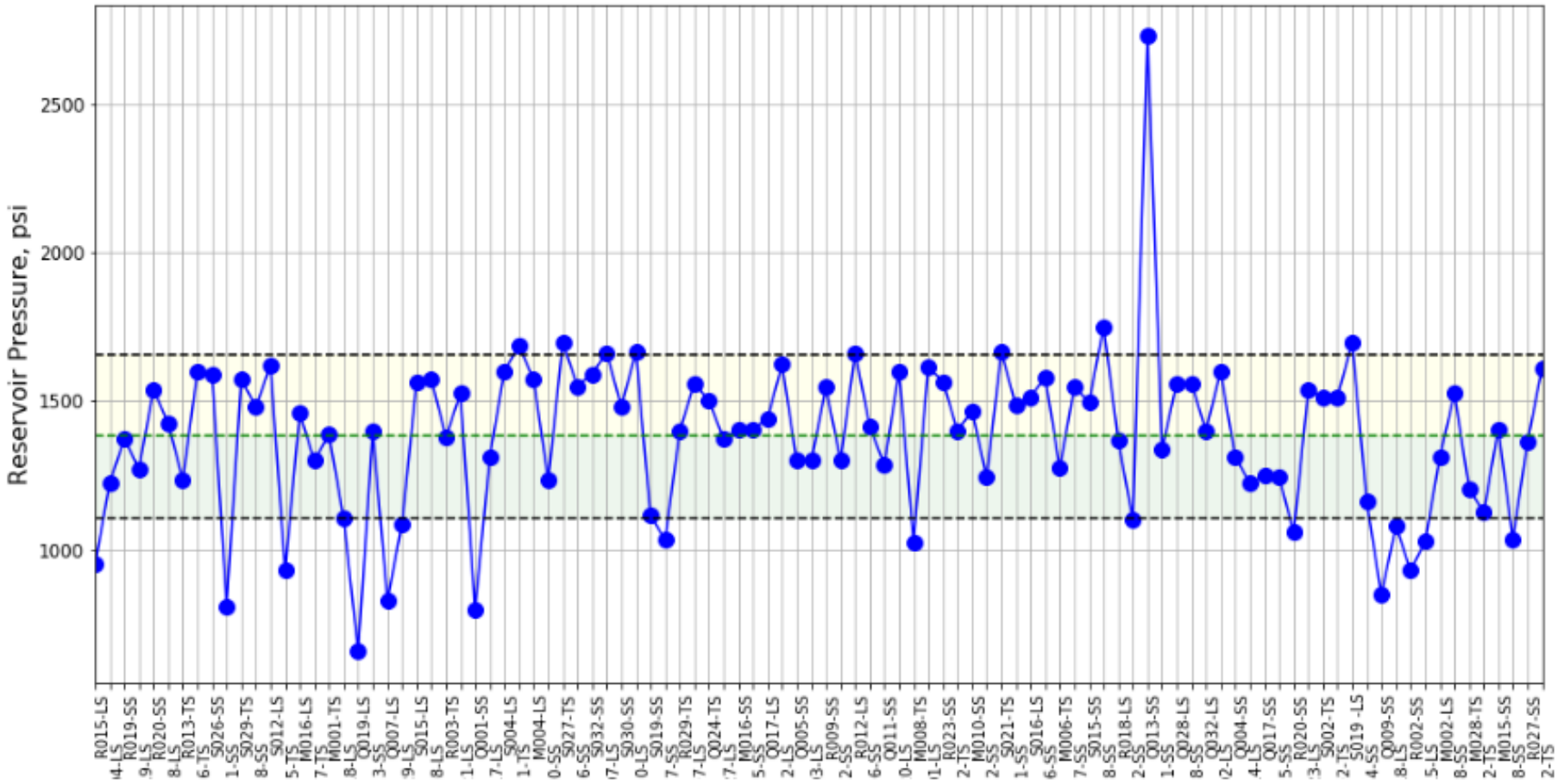
Detecting outliers

1. What are outliers
2. Should we delete outliers?
3. Methods for detecting outliers
4. Can you detect outliers with one well?

Example of outliers: RES_PRS, SD=1

```
## Using class method to plot Reservoir Pressure
```

```{python}
wds.plot_one_var_upper_lower('WT_RESPRES_On', 'Reservoir Pressure, psi', 'blue', nsd=1)
```



Example of outliers: RES_PRS, SD=3

```
python}
plot_one_var_upper_lower('WT_RESPRES_On', 'Reservoir Pressure, psi', 'blue', nsd=3)
```