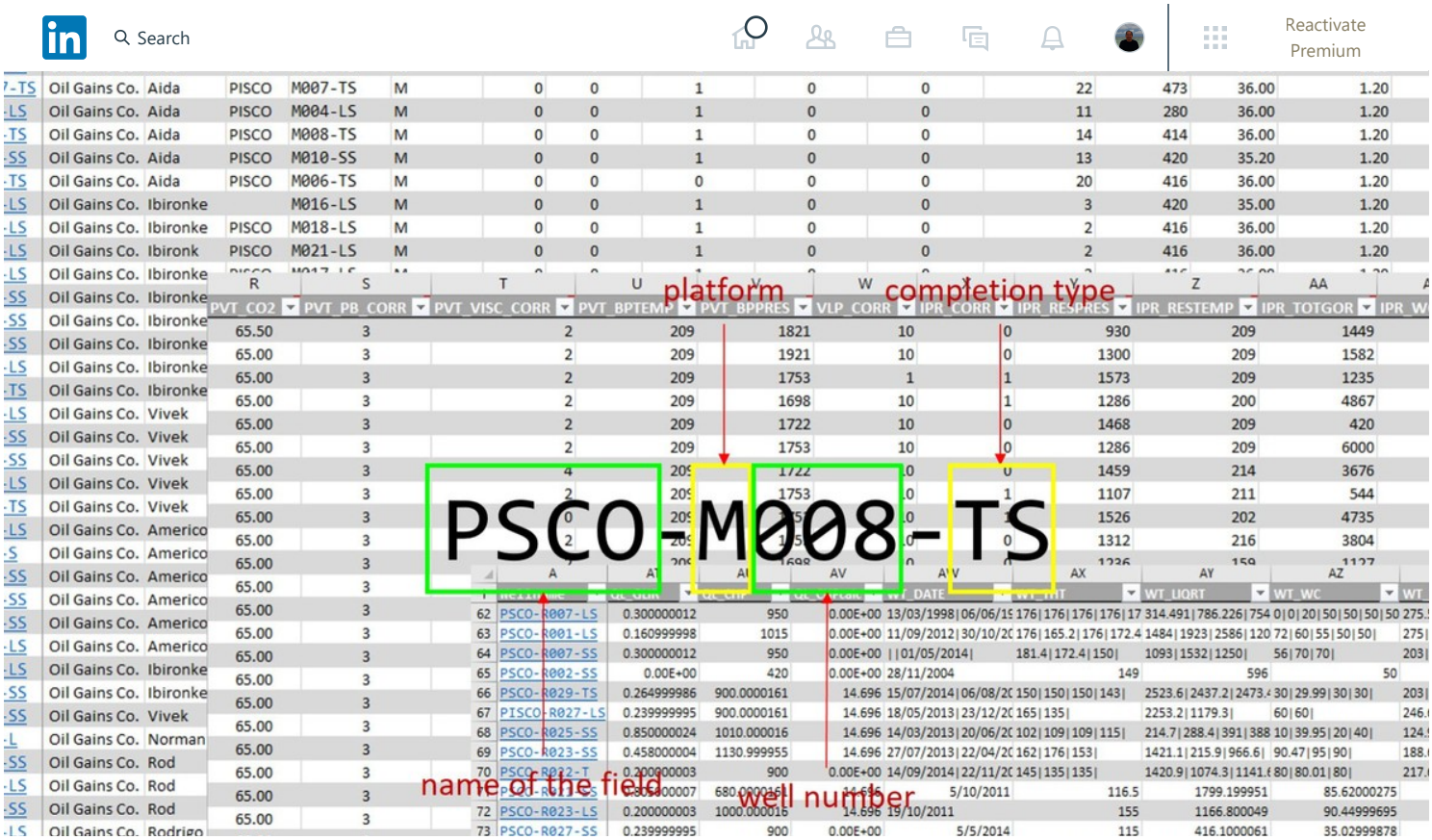


# Data Science for Petroleum Engineering - Part 5: "Transforming Excel well raw data into datasets."

Published on August 18, 2017 Edit article | View stats

Alfonso R. Reyes
Petroleum Engineering Data Scientist
32 articles

879 75 10 8

One of the big challenges of this new era of data science. machine learning and artificial intelligence is getting unhooked from the habit of working with spreadsheets. They have been around for 30+ years and were awesome. But spreadsheets - or worksheets - do not scale well with massive amounts of data; or continuous streams of data; or other characteristics that are key for taking good and sound decisions such as **reproducibility**. Besides, spreadsheets have not kept up with the times so we have seen the plotting capabilities getting very much behind of other software.

*Plots are the most expressive way that you can show your data and analysis.*

This time we will start with some well raw data. This data is part of the input data that we require to create well models for nodal analysis, production optimization, IPR/VLP calibration with well test data, troubleshooting, plan a stimulation job, or reviewing the well

same could have been used with Schlumberger's **Pipesim**, or any other.

Again, we will use R for these tasks. What we will do is:

1. Read the Excel data into R
2. Perform a basic statistics on the raw data
3. Find problems with data: data missing or improperly entered
4. Deal with missing data and correct typing issues
5. Convert the raw data to tidy data before analysis and plotting
6. Save the tidy data
7. See what story the data is trying to tell us
8. Present our discoveries

### Setting the stage

In order for you to be able to reproduce this analysis, you will need to install **R**, **Rtools** and **RStudio**. They are very easy to install. And the best of all, they are free.

Don't be mistaken. This is high quality software that will lead you to a world full of discoveries. So, I am assuming that at least you have installed R and that you already have your RStudio screen in front of you. This is supposed to be a sort of introductory session to R, so, I am assuming that you have little or no previous experience with R either. If you are an experienced user, you will skip to the end very quick.

Remember, R has been designed by scientists for the use of scientists and engineers. It is not only a tool for discovery but for development. I showed a little bit of it with the article on the compressibility factor.

### The Raw Data

We will start by reading the raw data. Raw data is data as-is. It hasn't been cleaned up or checked or organized. Although this raw data has had some treatment to allow us focus on the main goal. You will have access to the raw data via GitHub. I will publish all the material there: raw data, datasets, scripts, notebooks, etc. I may even publish a R **package** to make the installation much easier for you.

The raw data is to be used for input in 100 wells. This input data is the minimum required to create a well model under any nodal analysis software. The well data could be grouped as: general data (well name, field, platform), well type data (fluid, completion type, artificial lift

**General data:**

**Well type data:**

**PVT data:**

**IPR data:**

**Well test data:**

The well test data transformation into tidy data will be a major task but that's life. That's how raw data comes. And then we use tools like R for the data munging. It will be fun!

## Reading the raw data

Now, back to our RStudio screen. R can read virtually any data format out there. If you just installed R and haven't installed anything else what you have is **r-base**. You can do a lot of stuff with it. But you wouldn't able to read an Excel spreadsheet. You have to install a package for that. The packages are supplements to base R. If you need some specific type of plot or a statistical distribution that you didn't find in r-base you just install the package. There are 11,000+ of them. They can also be installed directly from the internet. We will start by installing the package **xlsx** which will allow us to read Excel .xlsx files.

```
> install.packages("xlsx")
```

Once the package is installed we proceed to read the raw data:

```
# read the raw data
myXl <- read.xlsx("./inst/extdata/oilfield_100w_raw_data.xlsx",
                  sheetIndex = 1, stringsAsFactors = FALSE)
```

save under **./data**.

The first part of the command we see **myXl**, which is an object that will be whatever the data is inside the file. read.xlsx is the function that reads the E comes the long string "./inst/extdata/oilfield_100w_raw_data.xlsx", then a number "1" that means the sheet number (sheetIndex).

After you run this command take a look the top right side of your screen. S Environment tab. You will see that the object **myXl** is showing this:

That means 100 **observations** or rows and 61 **variables** or columns. The raw data is already living in R. That is how rows and columns are called in data science jargon: observations and variables.

Now, if you double-click on the myXL object R automatically will open a data viewer for you.



You can get the raw data file oilfield_100w_raw_data.xlsx via GitHub. Download the file and start practicing opening the file and loading it in R.

### The notebook is your friend

Another thing that you will notice in this lecture is that we can combine text, math, equations and results in the same document. As a matter of fact, I am writing all of this in a R Markdown document or notebook. You can see it as the README of the package in GitHub here. It is the file README.md in green highlight.

Writing project or analysis documentation this way is not only useful but a time saver. You don't need to type your text in Word, for instance, and copy-paste the calculations or plots in the document afterwards. And most important of all, you **reduce the chance of errors**. You will see for yourself later when we mix calculations inside and together with the text.

### What's next?

- Data introspection
- Summary
- Finding and filling missing data

- Analysis and plotting of the numeric data
- Converting the well text data that is bar-separated to columnar format
- Join tables by a key variable

### Well naming convention

Before we begin some tips about the well naming that is used for classifica
this later for summarizing data such as how many wells per platform, what
completion has the best producers, what is the platform with wells with high watercut, etc.

As the figure explains, the **first four letters** is the abbreviated name of the **field**. Since we are working with one field only in this lecture, all of the wells should have the same field name. After the dash, next is the **platform**. It is only **one letter.** There are four platforms M, Q, R and S. They should be in uppercase. Next after that is the three (3) digit **well number**. Not four or two or one; it is **3-digit number**. Then a dash, and a **two-letter completion type**. Because we are using gas lift wells and have two producing zones we require dual completions, one with the long string (LS) and the other with the short string (SS). Wells with a unique tubing string are marked (TS).

If you are using the API 14-number well identifier, all this work still applies, with the obvious differences.

So, our first task is to ensure the wells are named correctly. That is essential for the classification and analysis that we will perform later.

Most likely what we are going to find is:

- Typos
- Combination of uppercase and lowercase
- Omitting the dashes; omitting letters
- Using arbitrary well numbers instead of 3-digit; or
- Absence of well name at all

We will address this using R.

Next, is **5.1 Data introspection**

Follow me in Twitter fonzie@oilgains

75 Likes

10 Comments

Show previous comments

**Burney Waring** • 1st
Director of Retirement Testing at Waring Retirement Laboratory

Thanks, **Alfonso R. Reyes**! This is a great example of teaching actual practical skills to network!

Like Reply | 1 Like · 1 Reply

**Alfonso R. Reyes** • You
Petroleum Engineering Data Scientist
1y

Thanks Burney. So much to do yet. I have new material in the pipeline: convolution for permeability in reservoir simulation, multiple VLP correlation in tubing, ordinary differential equations, etc. Once that's done, I will move to neural networks for correlations. Stay tuned.

Like Reply

**Danielle Sion-Moore** • 1st
Petroleum Engineer | Production Optimization | Well Operations | Completions | Artificial Lift
1y

I'm loving these Data Science for PE articles Alfonso! Great stuff!

Like Reply | 1 Like · 1 Reply

**Alfonso R. Reyes** • You
Petroleum Engineering Data Scientist
1y

Thanks Danielle.

Like Reply

Add a comment...

**Alfonso R. Reyes**
Petroleum Engineering Data Scientist

## More from Alfonso R. Reyes See all 32 articles

**A book review: Fundamentals of Data Visualization**
Alfonso R. Reyes on LinkedIn

**For what things R programming language is better than Python?**
Alfonso R. Reyes on LinkedIn

**An Artificial Lift Method Selector for Petroleum Engineering written in R**
Alfonso R. Reyes on LinkedIn

**Is R as versati**
Alfonso R. Reyes