Artificial Intelligence in Commodities Estimation for Construction Projects.

Ahmed Safwat Ewida, PMP®

Table of Contents

4. Lasso Regressor.
5. KerasRegressor.
6. RandomForestRegressor.
7. CatBoostRegressor.
8. LGBRegressor.
9. XGBRegressor.

4. RESULTS

1. Models Evaluation.
2. Justification and Conclusion

5. References

1. Introduction:

Artificial intelligence (AI) is a general term for describing when a machine mimics human cognitive functions, like problem-solving, pattern recognition, and learning. Machine learning is a subset of AI. Machine learning is a field of artificial intelligence that uses statistical techniques to give computer systems the ability to "learn" from big historical data, without being explicitly programmed. A machine becomes better at understanding and providing insights as it is exposed to more data.

1.1 Examples of Machine Learning and AI in Construction:

- **Prevent cost overruns**

Artificial Neural Networks are used on projects to predict cost overruns based on factors such as project size, contract type, crew composition, labours supervision, material availability, drawings availability, sub-contractors status, project location, political issues and the competence level of project teams and project managers. Historical data are used by predictive models to predict realistic timelines for future projects.

- **Risk Mitigation**

Every construction project has some risk that comes in many forms such as Quality, Safety, Time, and Cost Risk. The larger the project, the more risk, as there are multiple sub-contractors working on different trades in parallel on job sites. There are AI and machine learning solutions depending on Historical data can monitor ,prioritize and categorize risk and giving solutions to project managers to focus and work closely on high risk items.

- **AI for Construction Safety**

Construction workers are killed on the job five times more often than other labours. According to OSHA, the leading causes of private sector deaths (excluding highway collisions) in the construction industry were falls, followed by struck by an object, electrocution, and caught-in/between.

A typical construction project can have thousands of open issues, hundreds of RFIs, and numerous change orders that are open on any given day. Imagine a smart assistant who can analyse this mountain of project data and alert you about the top 10 critical things that need your attention today? Machine learning is that smart assistant, helping teams identify the most critical risk factors from a construction safety and quality perspective that need immediate attention.

Performing AI algorithms such as Real Time Object detection techniques can be used to identify and analyse safety hazards, categorize and tag site photography, and send notifications when PPE is not being properly used on the job site. It can even be used to identify who is violating safety standards, and tag them and/or their supervisors to address the problem.

- **Commodities Estimation and Productivity Improvement**

At a time when an overwhelming amount of data is being created every day, AI Systems are exposed to an endless amount of data to learn from and improve every day. This presents an opportunity for construction industry professionals to analyse and benefit from the insights generated from the data with the help of AI and machine learning systems.

Nowadays, Machine Learning and AI became the cornerstone in the estimation of Commodities and productivity / Norms such as :concrete, steel fixing , shuttering , piping fabrication , piping erection , equipment installation , steel structure fabrication /erection , Façade Systems , Finishing Works ,….etc.,our goal is to identify the features importance that impacting the productivity/Norms for different construction activities, and establishing standard rates for different commodities .

1.2 Problem Statement:

In this research, I have utilized the Machine Learning and AI techniques such as Artificial Neural networks, Boosting, regression and Natural Language Processing Techniques to predict the Piping spools Erection daily production and the features that impacting the piping erection productivity in industrial and oil and gas projects depending on historical data. The features which has been selected as listed below **(23 Features)**:

1-Country: The data has been collected for construction projects in oil and gas and petrochemical projects in five Countries: Egypt, Saudi Arabia, Qatar, UAE and Oman in the period from 2005 till 2017 with average number of projects 35 Projects as below:

- Egypt: average number of Projects 14 Projects.
- Saudi Arabia: average number of Projects 10 Projects.
- Qatar: average number of Projects 3 Projects.
- UAE: average number of Projects 4 Projects.
- Oman: average number of Projects 4 Projects

2-Drawings Availability: Two Categories have been selected of drawings availability (Low and High)

3-Fabricated Spools availability: Two Categories have been selected of drawings availability (Low and High)

4-Working at Heights: Two Categories have been selected of drawings availability (yes if the piping erection is on pipe racks with height more than 4.60 Meter , and otherwise NO)

5-HSE and Security Restrictions: Two Categories have been selected (yes in case of HSE and security restrictions and that has been found severely in Gulf countries and oil and gas life areas in Egypt , and otherwise NO)

6-Heat Index and Temperature: Two Categories have been selected (High in June, July and August in Gulf Countries, and otherwise Low)

7-Political issues: Two Categories have been selected (Yes, in financial crisis in 2008 and Arab spring (series of anti-government protests) in 2011, as many projects have been impacted, and otherwise No)

8-Crews Nationality: Divided between Two Categories (Arab and others)

9-Number of Pipe Fitters.

10-Number of Argon Welders.

11-Number of CS Welders.

12-Number of cranes.

13-Number of Riggers.

14-Number of Grinders.

15-Holidays: Two Categories have been selected (Yes and No)

16- Distance between the spools fabrication Workshop and site (Low in case of the distance is within the site boundary, and otherwise High)

17-Crew Experience: Two Categories have been selected (Low in case of the average experience is less than 5 years and, otherwise High)

18-Material of Pipes: Four Categories have been selected (Carbon Steel, Stainless steel, Low Temp, Duplex)

19-Pipes Diameter: Three Categories have been selected (Low if the pipes Diameter less than 10 D.I, Medium if Diameter between 10 and 22 D.I and Large if the Diameter More than 22 D.I)

20-Material Availability: Two Categories have been selected (Yes and No)

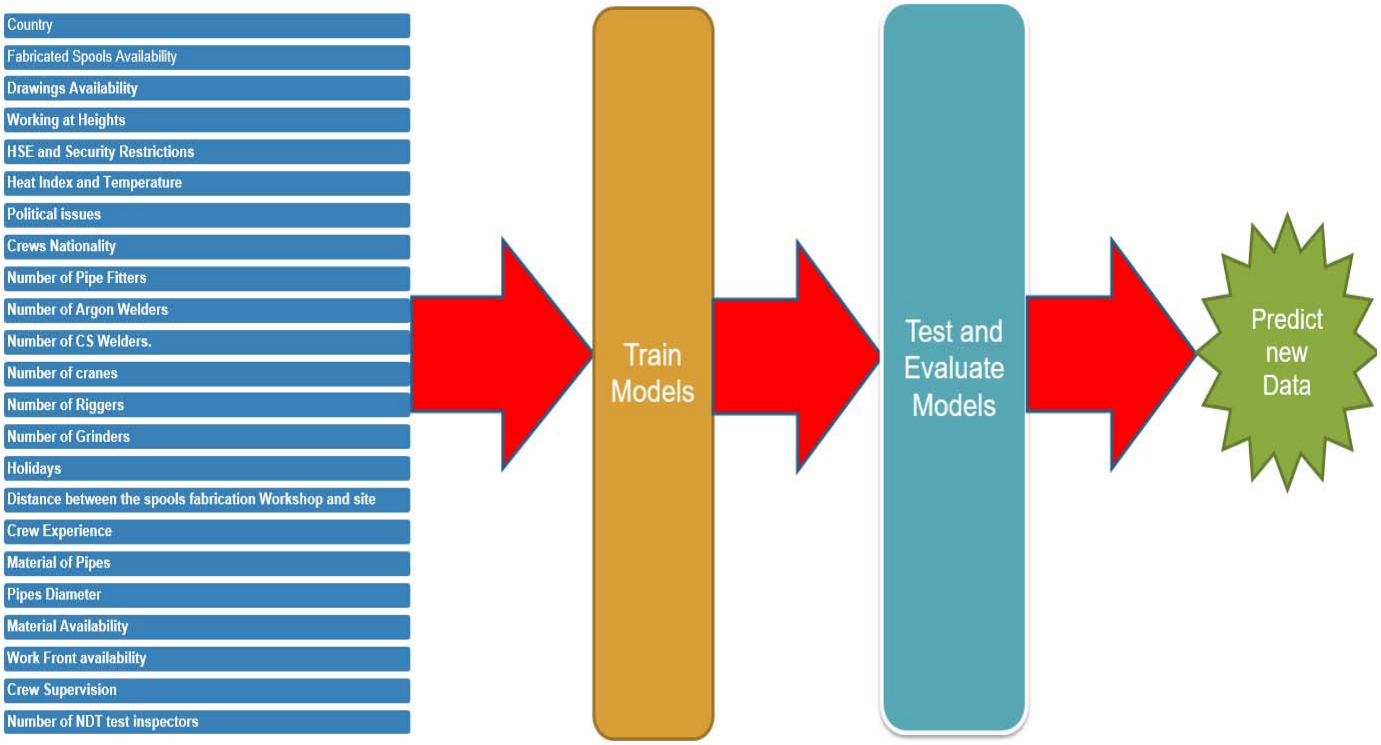
21-Number of NDT test inspectors.

22-Crew Supervision: Two Categories have been selected (Normal if the ratio of Supervision to direct labours is within range 10 % to 15% and low if it is less than 10 %, and otherwise high)

23-Work Front availability: Two Categories have been selected (Yes and No)

Now, we will study the impact of these features on the Piping erection production, then we want to train and test our models with the data we have, and then obtain the best model that can predict our daily production.

The Below figure showing our Process from Features selection till predicting the data passing by training the Models, testing and Evaluation.



1.3 Solution statement:

We will Follow the below process in our Problem Solution:

NOTE: The Programming Language used in the research is **Python(3.7)**

- **Fetching the Data:**

  The data has been collected from Daily reports, which encompasses to the daily production of piping erection, manpower status, Equipment status, Heat index/Temperature and reasons of delay.

- **Clean /preparation Data:**

1. Wrangle data and prepare it for training.
2. Using web scarping to collect the official holidays in Egypt, Qatar, Saudi Arabia ,UAE and Oman for the period from 2005 till 2017
3. Remove Duplicates and outliers and dealing with missing data, convert categorical data, normalizing the float and integer values.
4. Using Natural Language Processing Techniques to collect the area of Concern from daily reports which impacting the production rate.

- **Data Visualizing and analysis:**

1. Visualize data to help detect relevant relationships between variables.
2. Split into training and evaluation sets

- **Taring Model:**

  The goal of training is to make a prediction correctly as often as possible, the model becomes better as it is trained to more data.

- **Evaluating the Model:**

1. Using some metric or combination of metrics to measure the performance of model.
2. Shuffling the data and selecting 15/85 ratio for test/train data set.
3. Hyper-parameter tuning, which is a corner stone for Model efficiency and performance improvement.
4. Using test set data which have to predict the output.

1.4 Evaluation Metrics:

Our Problem is Regression Problem that will lead us to use the following Metrics:

Root Mean Squared Error: Root mean squared error (RMSE) is the square root of the mean of the square of all of the error. The use of RMSE is very common, and it is considered an excellent general-purpose error metric for numerical predictions.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(S_i - O_i)^2}$$

Where *Oi* are the observations, *Si* predicted values of a variable, and *n* the number of observations available for analysis. RMSE is a good measure of accuracy, but only to compare prediction errors of different models or model configurations for a particular variable and not between variables, as it is scale-dependent.

2.Data Analysis:

2.1 Data Exploration:

We will dig more and more in our data and make our statistics to see what the nature of our historical data is:

```
piping_df.shape
```

```
(158868, 29)
```

The Shape of our Data is **15,868 rows (inputs) and 29 Columns (Features) with total 460,172 records**

The Features name and type are as below:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 158868 entries, 0 to 158867
Data columns (total 29 columns):
Day                                                    158868 non-null datetime64[ns]
Country1                                               158868 non-null object
Project                                                158868 non-null object
HSE restrictions                                       158868 non-null object
Temperature/Heat index                                 158868 non-null object
Political issues                                       158868 non-null object
Material of Pipes                                      158868 non-null object
Pipes size in D.I                                      158868 non-null object
availability of Material                               158868 non-null object
Holiday                                                158868 non-null object
Pipe Fitter                                            158868 non-null float64
Grinder                                                158868 non-null float64
Pipe Welder (CS)                                       158868 non-null float64
Pipe Welder (Argon)                                    158868 non-null float64
Riggers                                                158868 non-null float64
Cranes                                                 158868 non-null float64
Fabricated Spools availability                         158868 non-null object
No of Inspectors for NDT Test                          158868 non-null object
Distance between the spools fabrication Workshop and site(L.M)    158868 non-null object
Crews Nationality                                      158868 non-null object
Crew Experience                                        158868 non-null object
crew supervision                                       158868 non-null object
Drawings availability                                  158868 non-null object
Work Front availability                                158868 non-null object
Working at heights                                     158868 non-null object
Erection                                               158868 non-null float64
Month                                                  158868 non-null float64
Year                                                   158868 non-null float64
Daily_Rate                                             158868 non-null float64
dtypes: datetime64[ns](1), float64(10), object(18)
```

We will drop some features like project name, Month, Year, Day as it will be misleading in model training.

Our Target will be the Erection Column.

The below table showing some statistics about our data:

	Pipe Fitter	Grinder	Pipe Welder (CS)	Pipe Welder (Argon)	Riggers	Cranes	Erection	Month	Year	Daily_Rate
count	158868.000000	158868.000000	158868.000000	158868.000000	158868.000000	158868.000000	158868.000000	158868.000000	158868.000000	158868.000000
mean	271.405595	746.256087	339.821493	202.829254	746.717501	135.702797	6938.930949	6.523378	2010.572796	24.935513
std	140.639659	435.763657	220.667145	273.841275	436.563000	70.319829	4700.511518	3.448591	3.493172	7.377627
min	30.000000	60.000000	30.000000	0.000000	60.000000	15.000000	465.000000	1.000000	2005.000000	11.500000
25%	153.000000	398.000000	175.000000	0.000000	396.000000	76.500000	3421.600000	4.000000	2008.000000	19.600000
50%	263.000000	681.000000	300.000000	0.000000	682.000000	131.500000	5902.400000	7.000000	2010.000000	23.100000
75%	375.000000	1016.000000	444.000000	348.000000	1016.000000	187.500000	9216.000000	10.000000	2013.000000	28.700000
max	599.000000	2396.000000	1198.000000	1198.000000	2396.000000	299.500000	32945.000000	12.000000	2017.000000	55.000000

The Maximum piping Erection per Day is **32945 D.I/Day.**

The Minimum piping Erection per Day is **466 D.I/Day.**

The Mean piping Erection per Day is **6938 D.I/Day.**

The 75% of the records have piping Erection **9216 D.I/Day.**

The below table showing first five rows in our dataset:

Day	Country1	Project	HSE restrictions	Temperature/Heat index	Political issues	Material of Pipes	Pipes size in D.I	availability of Material	Holiday	Pipe Fitter	Grinder	Pipe Welder (CS)	Pipe Welder (Argon)	Riggers	Cranes	Fabricated Spools availability	No of Inspectors for NDT Test	Distance between the spools fabrication Workshop and site(L.M)	Crews Nationality	Crew Experience	crew supervision	Drawings availability	Work Front availability	Working at heights	Erection	Month	Year	Daily_Rate
1/1/05	Egypt	project	NO	Low	NO	CS	Med	YES	No	67	134	67	0	201	33.5	High	High	Low	Arab	High	High	Low	High	NO	2110.5	1	2005	31.5
2/1/05	Egypt	project	NO	Low	NO	SS	Med	YES	No	265	530	265	265	530	132.5	High	High	Low	others	High	High	High	High	NO	10971	1	2005	41.4
3/1/05	Egypt	project	NO	Low	NO	LT	Med	YES	No	102	306	102	102	306	51	High	Low	Low	others	High	High	High	High	NO	4773.6	1	2005	46.8
4/1/05	Egypt	project	NO	Low	NO	Duplex	Low	NO	No	174	522	174	174	348	87	High	Med	Low	others	High	High	High	Low	NO	4089	1	2005	23.5
5/1/05	Egypt	project	NO	Low	NO	CS	Med	NO	No	159	477	159	0	477	79.5	High	Med	Low	others	High	High	High	High	NO	4293	1	2005	27

As seen above a sample of our data exploring Erection production in Egypt in 2005

Day	Country1	Project	HSE restrictions	Temperature/Heat index	Political issues	Material of Pipes	Pipes size in D.I	availability of Material	Holiday	Pipe Fitter	Grinder	Pipe Welder (CS)	Pipe Welder (Argon)	Riggers	Cranes	Fabricated Spools availability	No of Inspectors for NDT Test	Distance between the spools fabrication Workshop and site(L.M)	Crews Nationality	Crew Experience	crew supervision	Drawings availability	Work Front availability	Working at heights	Erection	Month	Year	Daily_Rate
1/1/05	Qatar	project	YES	Low	NO	CS	Med	YES	No	122	366	122	0	244	61	High	Low	Low	others	High	High	High	Low	NO	3416	1	2005	28
2/1/05	Qatar	project	YES	Low	NO	Duplex	Med	NO	No	69	138	69	138	207	34.5	High	High	Low	others	High	High	High	High	NO	1035	1	2005	15
3/1/05	Qatar	project	YES	Low	NO	CS	Med	YES	No	53	212	106	0	159	26.5	High	High	High	others	High	High	High	Low	NO	1590	1	2005	30
4/1/05	Qatar	project	YES	Low	NO	LT	Med	NO	No	375	750	375	750	750	187.5	High	Med	Low	others	High	High	High	High	NO	5250	1	2005	14
5/1/05	Qatar	project	YES	Low	NO	CS	Med	YES	No	329	658	329	0	987	164.5	Low	Med	Low	Arab	High	High	High	High	YES	6678.7	1	2005	20.3

As seen above a sample of our data exploring Erection production in Qatar in 2005

Day	Country1	Project	HSE restrictions	Temperature/Heat index	Political issues	Material of Pipes	Pipes size in D.I	availability of Material	Holiday	Pipe Fitter	Grinder	Pipe Welder (CS)	Pipe Welder (Argon)	Riggers	Cranes	Fabricated Spools availability	No of Inspectors for NDT Test	Distance between the spools fabrication Workshop and site(L.M)	Crews Nationality	Crew Experience	crew supervision	Drawings availability	Work Front availability	Working at heights	Erection	Month	Year	Daily_Rate
27/12/15	Saudi	project	YES	Low	NO	CS	Med	NO	No	233	466	233	0	466	116.5	High	Low	Low	Arab	High	High	Low	High	NO	5382.3	12	2015	23.1
28/12/15	Saudi	project	YES	Low	NO	CS	High	YES	No	203	609	203	0	609	101.5	High	Med	Low	Arab	Low	High	High	High	NO	3410.4	12	2015	16.8
29/12/15	Saudi	project	YES	Low	NO	CS	High	YES	No	341	1023	682	0	1364	170.5	High	Med	Low	others	Low	Low	High	Low	NO	6547.2	12	2015	19.2
30/12/15	Saudi	project	YES	Low	NO	LT	Med	NO	No	490	1470	490	980	980	245	High	Med	Low	others	High	High	Low	Low	NO	10290	12	2015	21
31/12/15	Saudi	project	YES	Low	NO	CS	Med	YES	No	134	268	134	0	402	67	High	Med	Low	others	Low	High	High	High	NO	4556	12	2015	34

As seen above a sample of our data exploring Erection production in Saudi Arabia in 2015

Day	Country1	Project	HSE restrictions	Temperature/Heat index	Political issues	Material of Pipes	Pipes size in D.I	availability of Material	Holiday	Pipe Fitter	Grinder	Pipe Welder (CS)	Pipe Welder (Argon)	Riggers	Cranes	Fabricated Spools availability	No of Inspectors for NDT Test	Distance between the spools fabrication Workshop and site(L.M)	Crews Nationality	Crew Experience	crew supervision	Drawings availability	Work Front availability	Working at heights	Erection	Month	Year	Daily_Rate
27/12/11	Oman	project	YES	Low	YES	CS	High	YES	No	143	286	143	0	429	71.5	High	Med	Low	others	Low	Low	High	Low	YES	3324.75	12	2011	23.25
28/12/11	Oman	project	YES	Low	YES	CS	Med	NO	No	158	316	158	0	316	79	High	Med	Low	others	Low	High	High	High	NO	3792	12	2011	24
29/12/11	Oman	project	YES	Low	YES	SS	High	YES	No	118	236	118	118	354	59	High	Med	Low	Arab	High	High	Low	High	NO	2566.5	12	2011	21.75
30/12/11	Oman	project	YES	Low	YES	CS	Med	NO	No	225	450	225	0	450	112.5	High	Med	Low	others	High	High	Low	High	YES	4556.25	12	2011	20.25
31/12/11	Oman	project	YES	Low	YES	Duplex	Med	YES	No	249	747	249	498	498	124.5	High	Med	Low	others	High	High	Low	Low	YES	5789.25	12	2011	23.25

As seen above a sample of our data exploring Erection production in Oman in 2011

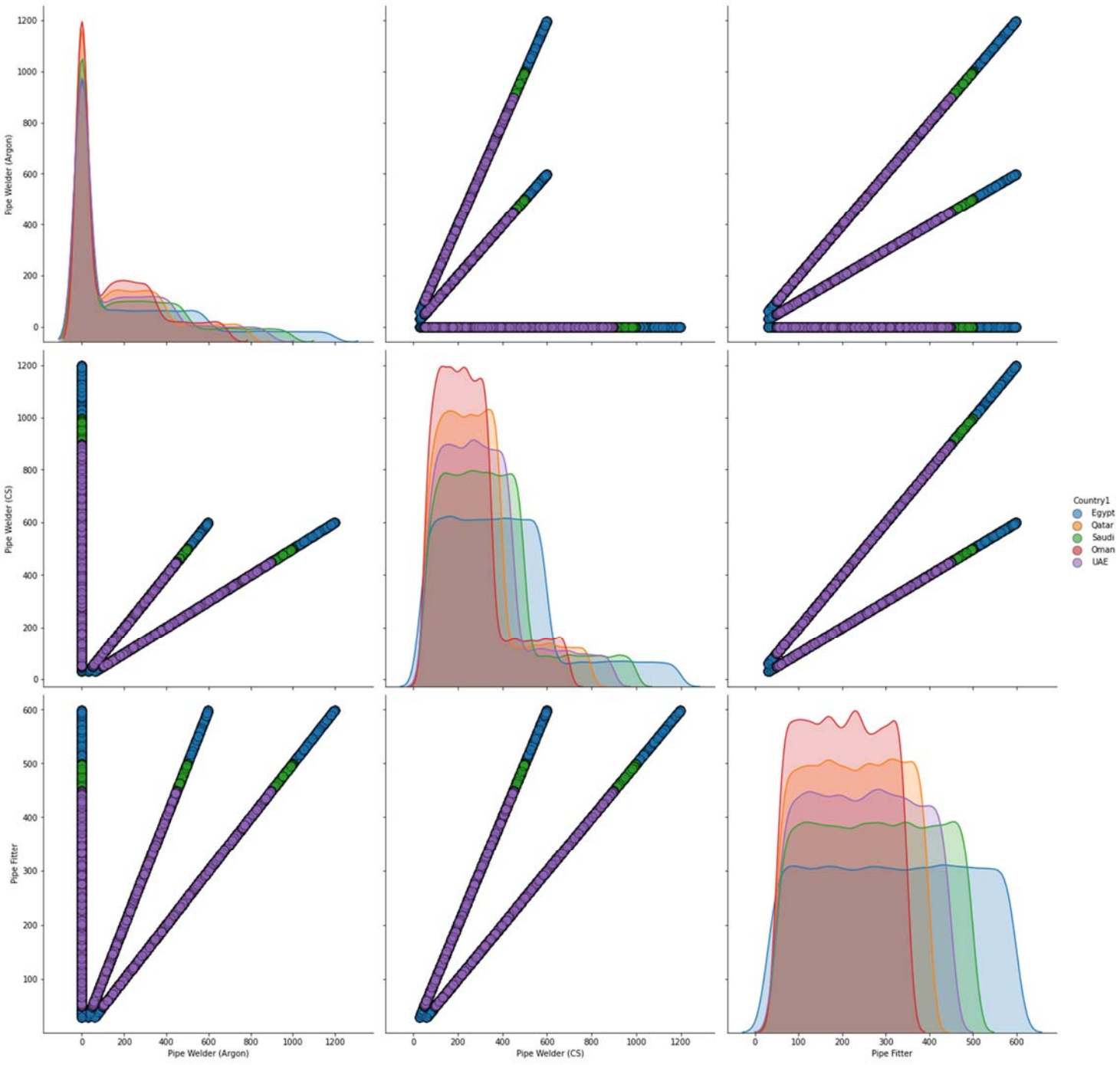
Day	Country1	Project	HSE restrictions	Temperature/Heat index	Political issues	Material of Pipes	Pipes size in D.I	availability of Material	Holiday	Pipe Fitter	Grinder	Pipe Welder (CS)	Pipe Welder (Argon)	Riggers	Cranes	Fabricated Spools availability	No of Inspectors for NDT Test	Distance between the spools fabrication Workshop and site(L.M)	Crews Nationality	Crew Experience	crew supervision	Drawings availability	Work Front availability	Working at heights	Erection	Month	Year	Daily_Rate
27/12/17	UAE	project	YES	Low	NO	LT	Med	YES	No	98	196	98	98	196	49	High	Low	Low	others	High	High	High	High	NO	2116.8	12	2017	21.6
28/12/17	UAE	project	YES	Low	NO	CS	Med	YES	No	416	1664	832	0	1664	208	Low	Low	Low	others	Low	Low	High	High	NO	7280	12	2017	17.5
29/12/17	UAE	project	YES	Low	NO	CS	Low	YES	No	115	345	115	0	345	57.5	High	Med	Low	Arab	High	High	Low	Low	YES	2012.5	12	2017	17.5
30/12/17	UAE	project	YES	Low	NO	CS	Med	YES	No	137	411	274	0	411	68.5	High	Low	High	others	High	High	High	Low	NO	3288	12	2017	24
31/12/17	UAE	project	YES	Low	NO	SS	Med	NO	No	205	615	205	410	410	102.5	High	Med	Low	others	High	High	High	High	NO	2870	12	2017	14

As seen above a sample of our data exploring Erection production in UAE in 2017

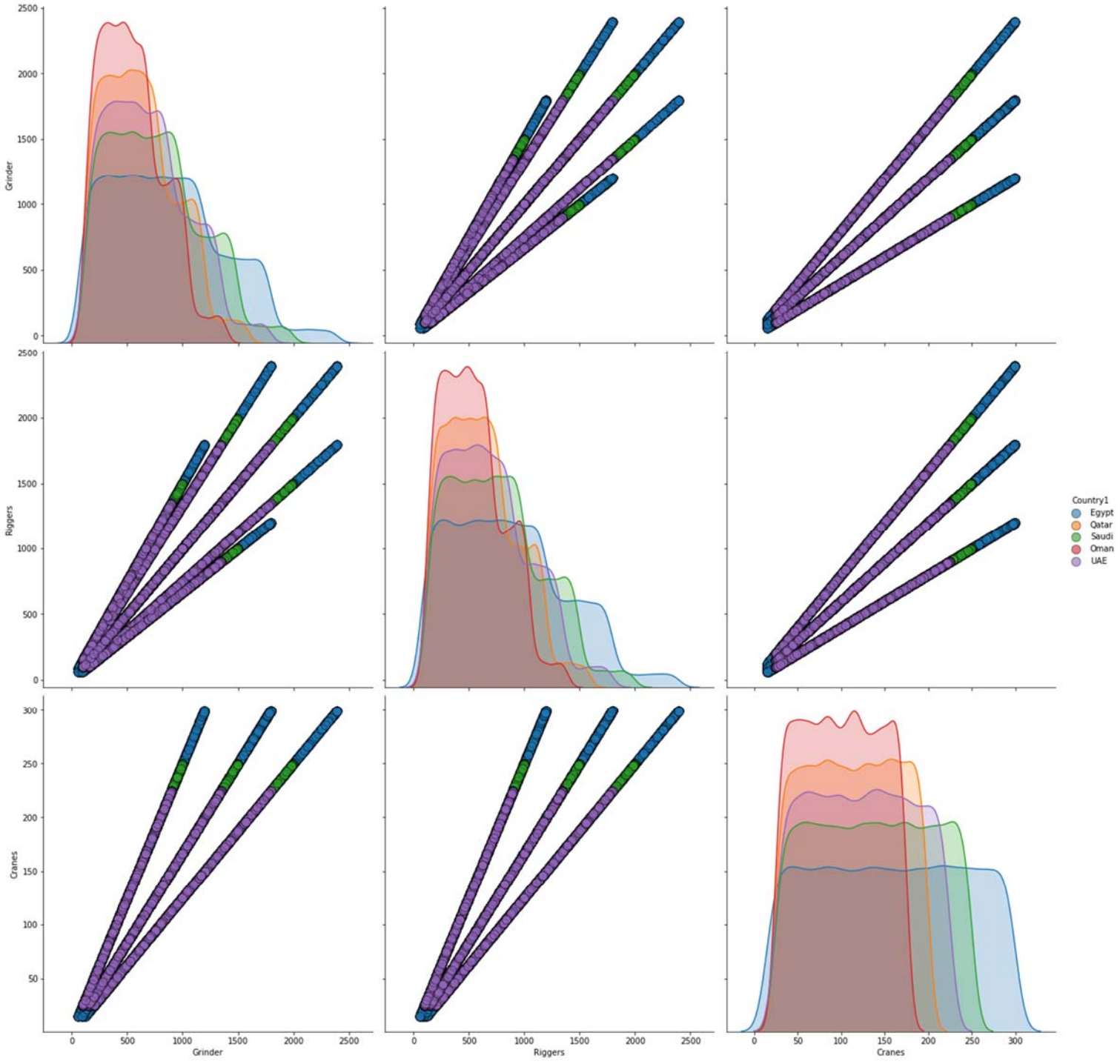
2.2 Exploratory Visualization:

Pair Plot between pipe fitter, Argon Welder, C.S Welders per Country

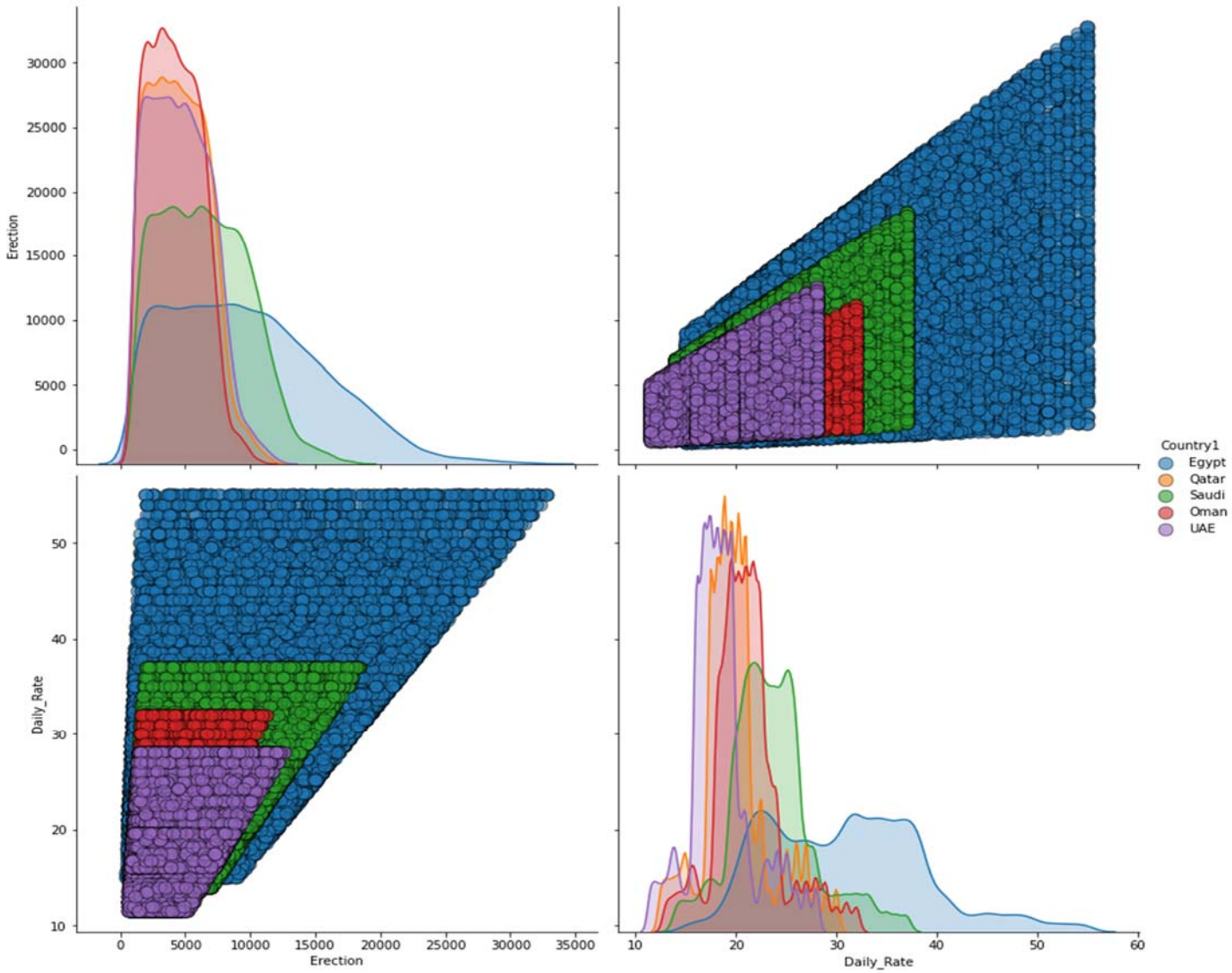
The Pair plot will create a grid of Axes such that each numeric variable in `data` will by shared in the y-axis across a single row and in the x-axis across a single column. The diagonal Axes are treated differently, drawing a plot to show the univariate distribution of the data for the variable in that column.



Pair Plot between Cranes, Riggers, Grinders per Country



Pair Plot between Cranes, Riggers, Grinders per Country

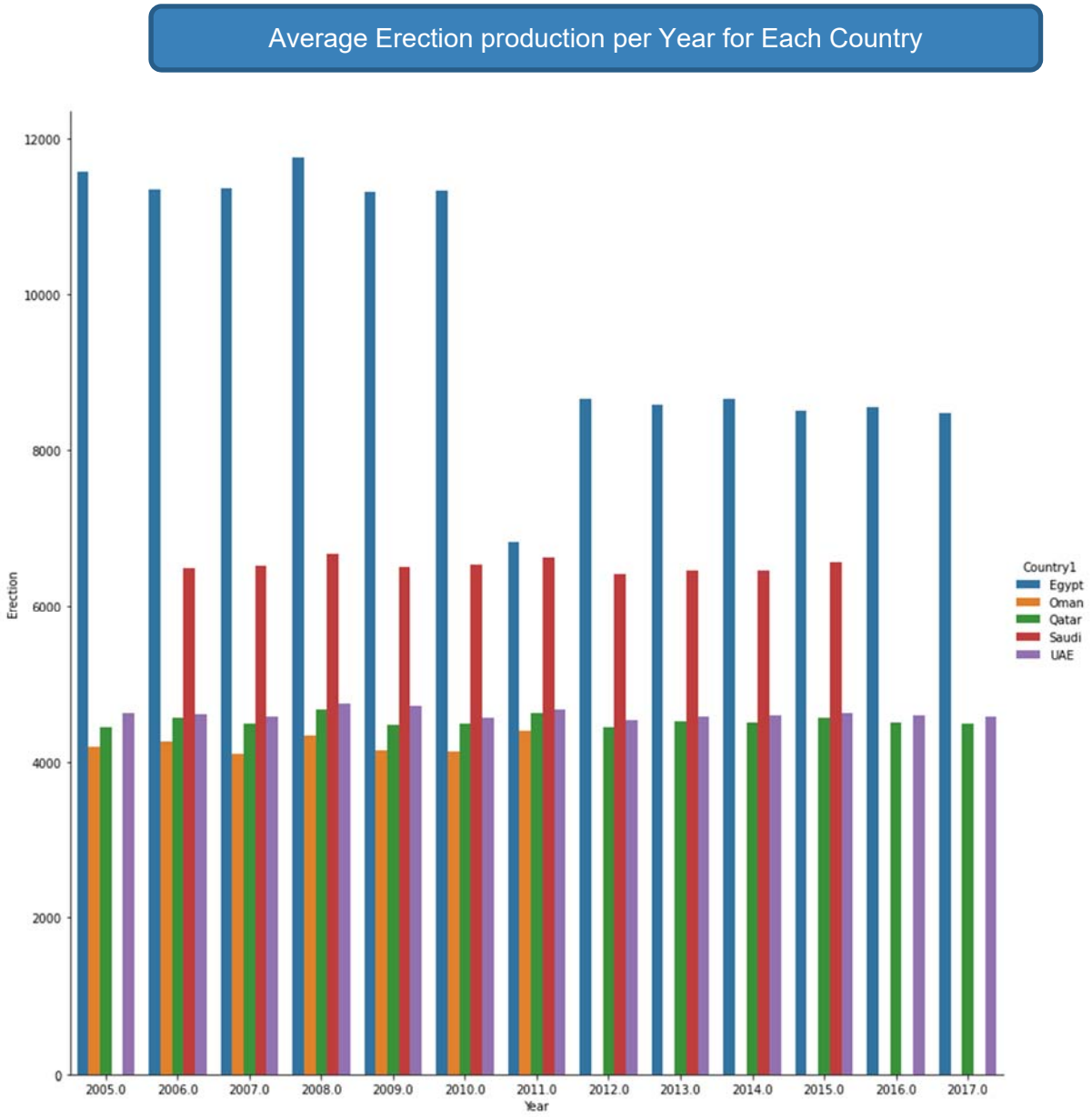


Obviously, from the above pair plot figure and the Gaussian Distribution theirs is a big standard deviation for the Erection records in Egypt, which vary from Zero to 35,000 .The Erection production in Egypt which is more than 20,000 D.I/Days is around 5% from the overall records of Egypt, and the production more than 10,000 D.I/Day is around 45% and the maximum is 32,945 D.I/Day, and minimum is 465 D.I/Day, and average 9,775 D.I/day.

Regarding the Daily rate of Erection, we can obtain the same results, Egypt has a big standard Deviation with average per Crew is 31 D.I/Day and Maximum 55 D.I/Day and Minimum 15 D.I/Day

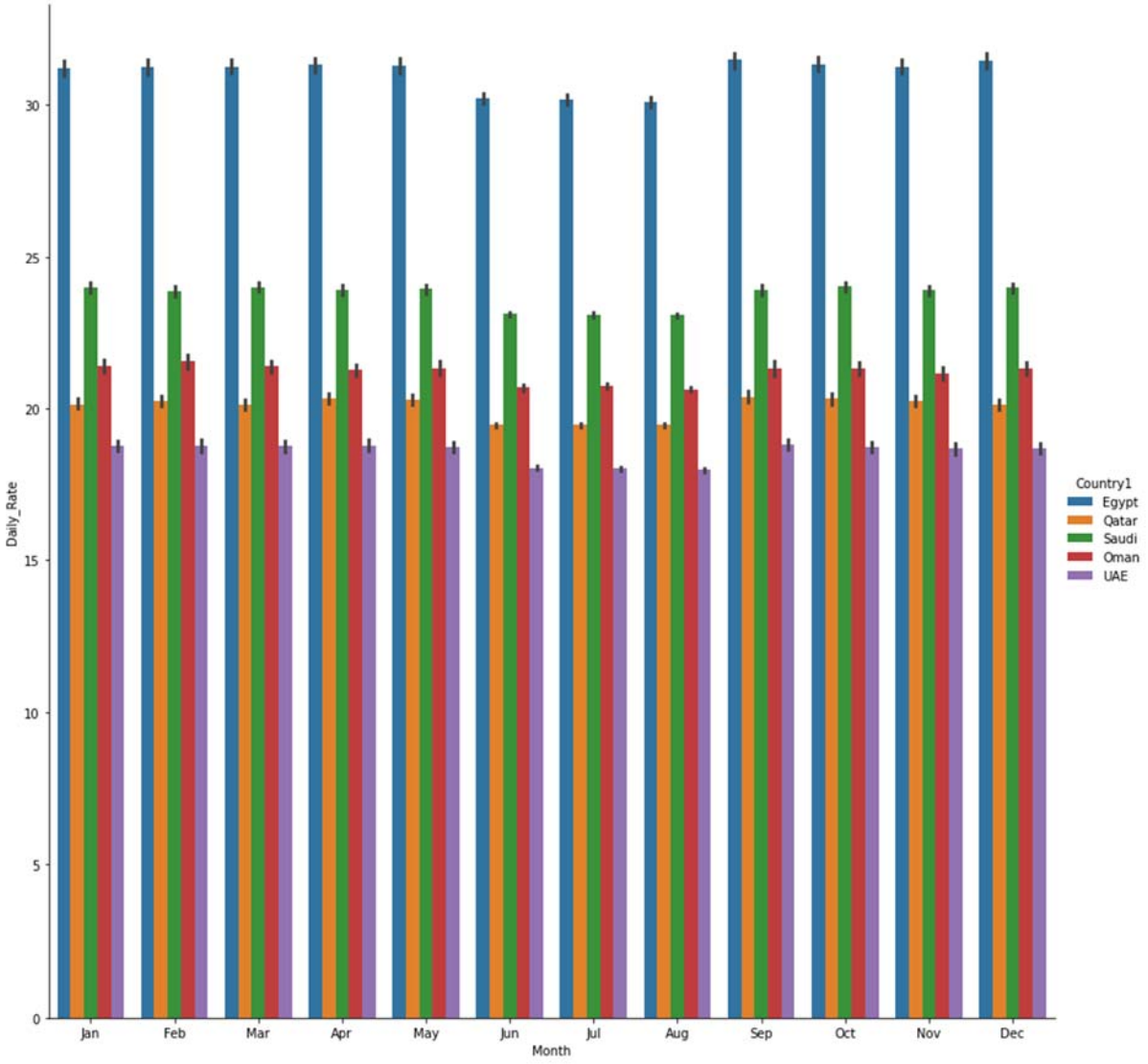
On the Contrary, the Gaussian Distribution in Gulf countries has small standard Deviation, which vary from 0 to 18,000 .The Erection production which is more than 10,000 D.I/Days is around 13% from the overall records of Gulf countries, and the production more than 5,000 D.I/Day is around 89% and the Maximum is 18,463 D.I/Day and the minimum is 610 D.I /Day, and average 5,352 D.I/Day

Regarding the Daily rate of Erection, we can obtain the same results, gulf Countries has a big standard Deviation with average per Crew is 22 D.I/Day and Maximum 37 D.I/Day, and Minimum 11.5 D.I/day



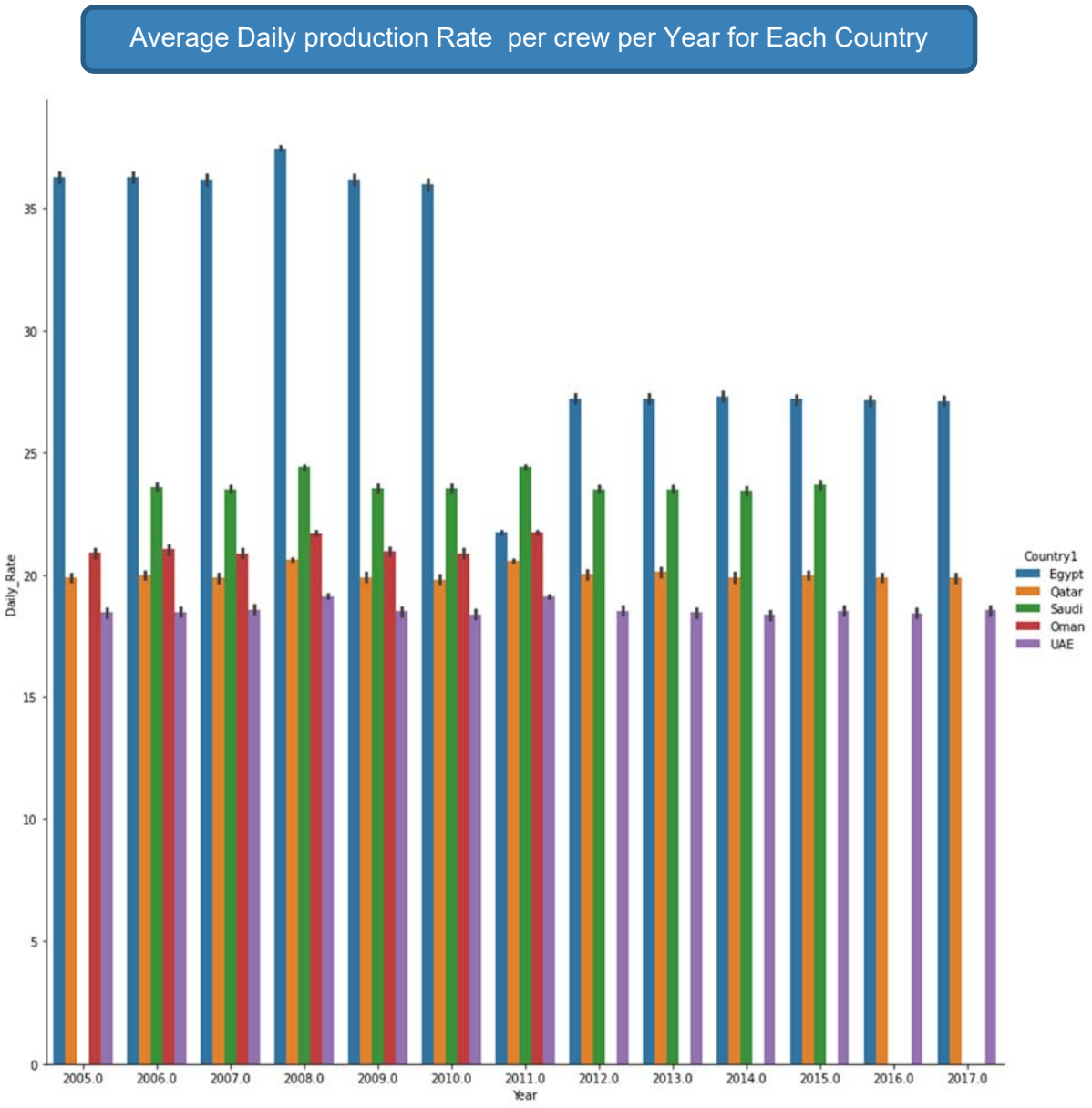
Obviously, there is a drop in production in piping erection in Egypt starting from 2011 till 2017, although the average number of the projects are almost same within that period.

Average Daily production Rate per crew per Year for Each Country



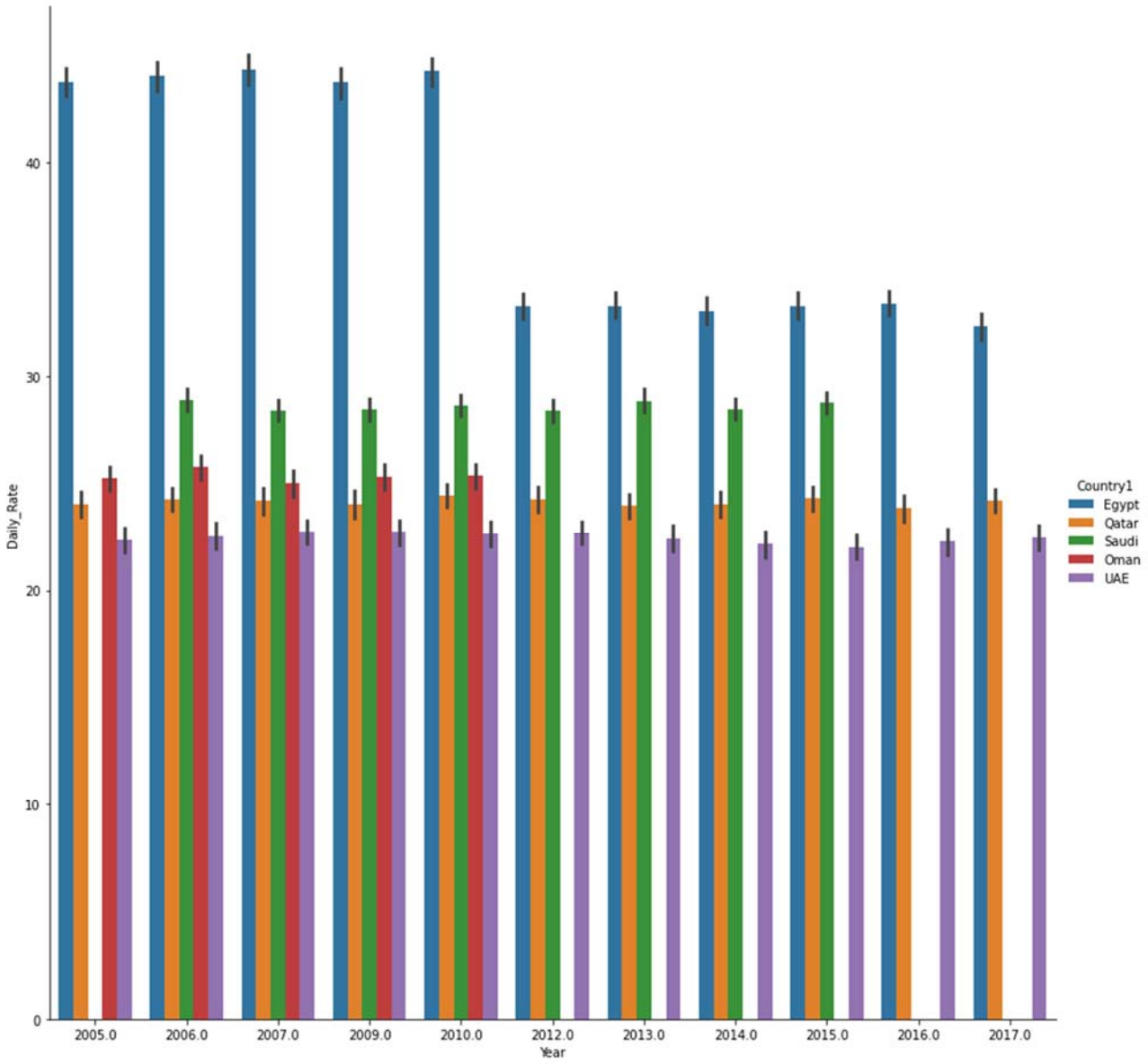
As seen in the above mentioned figure , the average daily production rate per crew in Egypt is around 31 D.I /Crew , Oman 21 D.I/Crew , Saudi Arabia 23.5 D.I/Crew , Qatar 20 D.I /Crew and UAE 18.5 D.I /Crew .

Obviously, The Crew Productivity has been dropped in summer season in all countries.

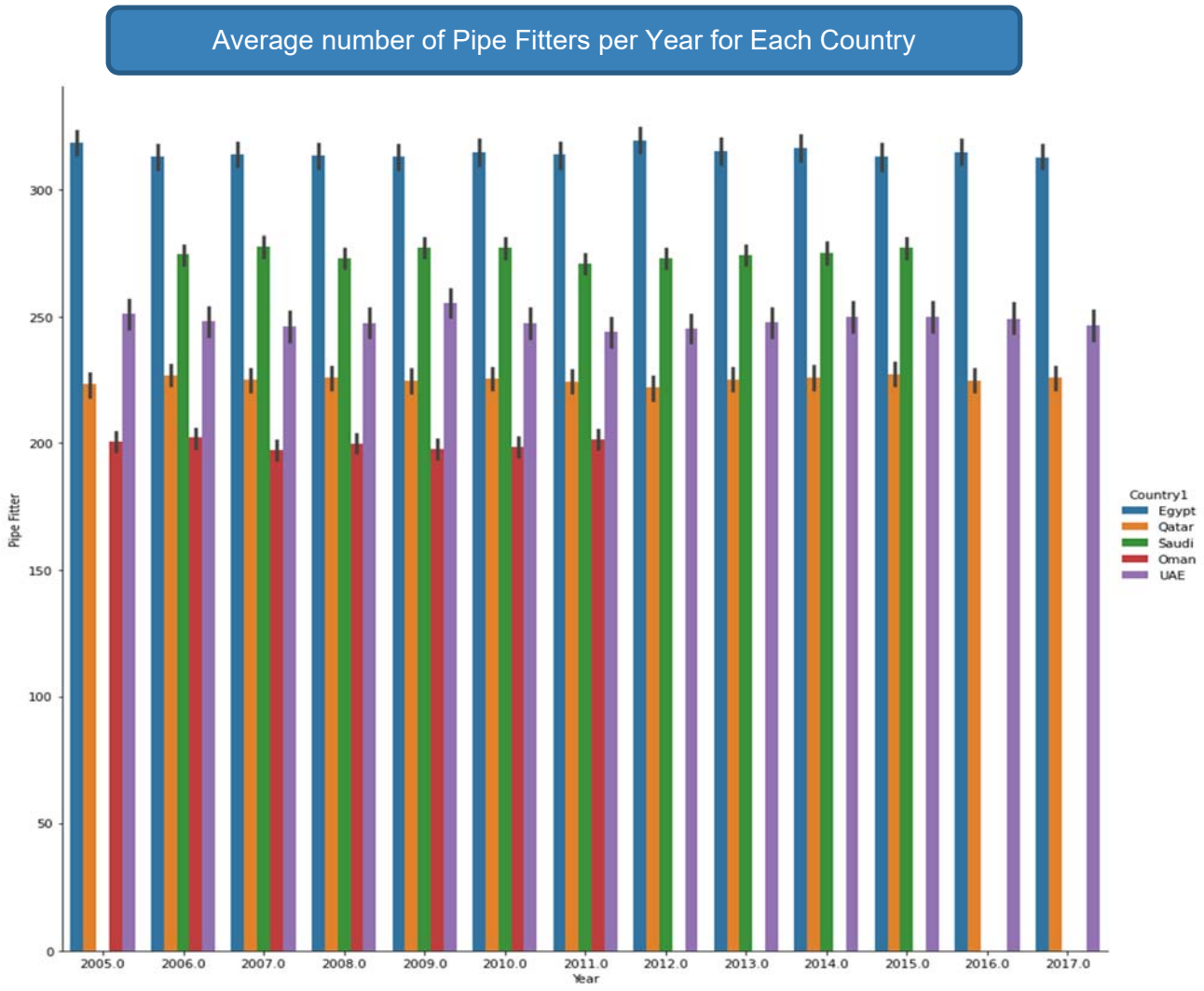


Obviously, The Crew Productivity has been dropped in Egypt starting from 2011 till 2017.

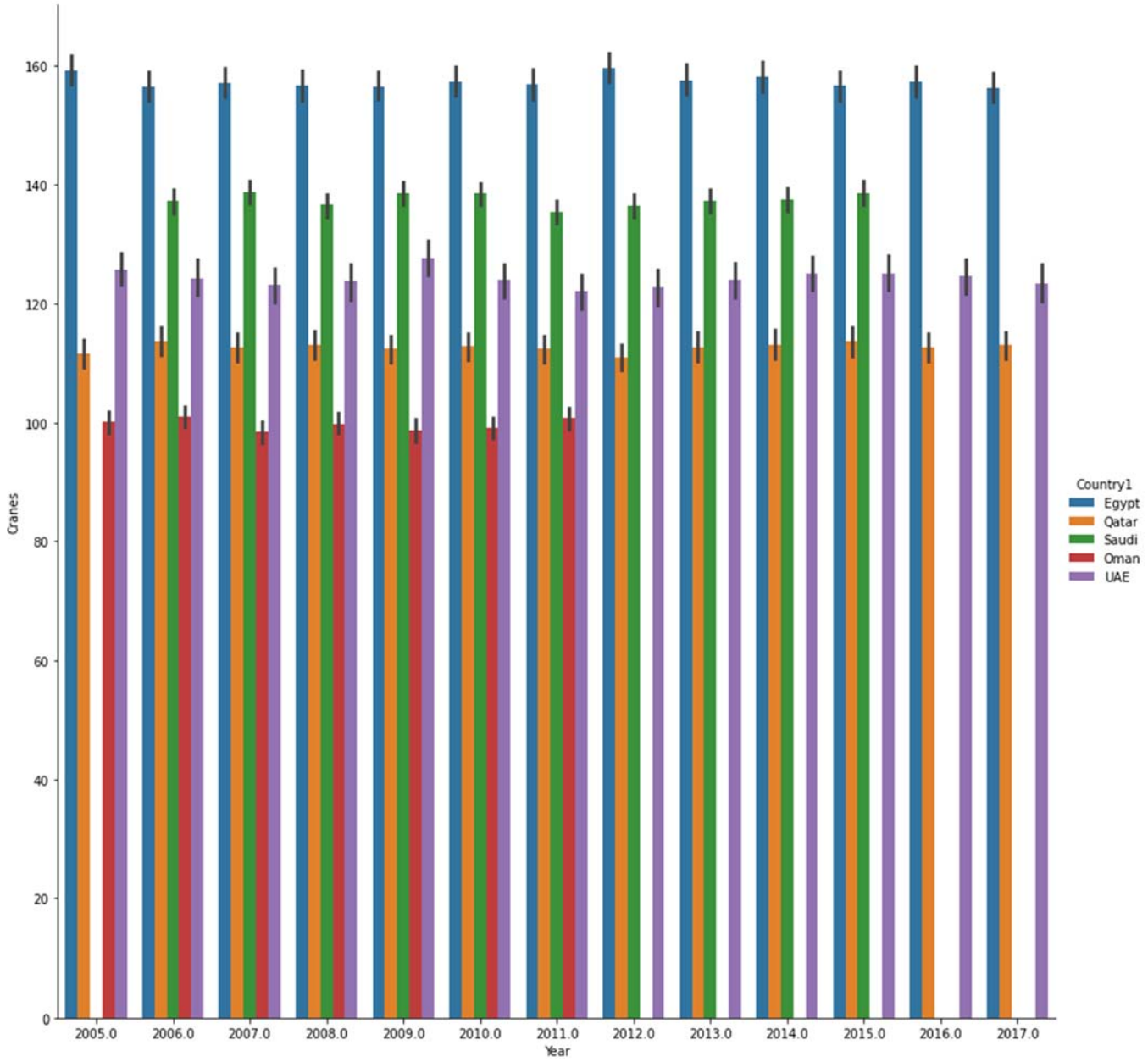
Average Daily production Rate per crew per Year for Each Country Excluding Unusual Circumstances



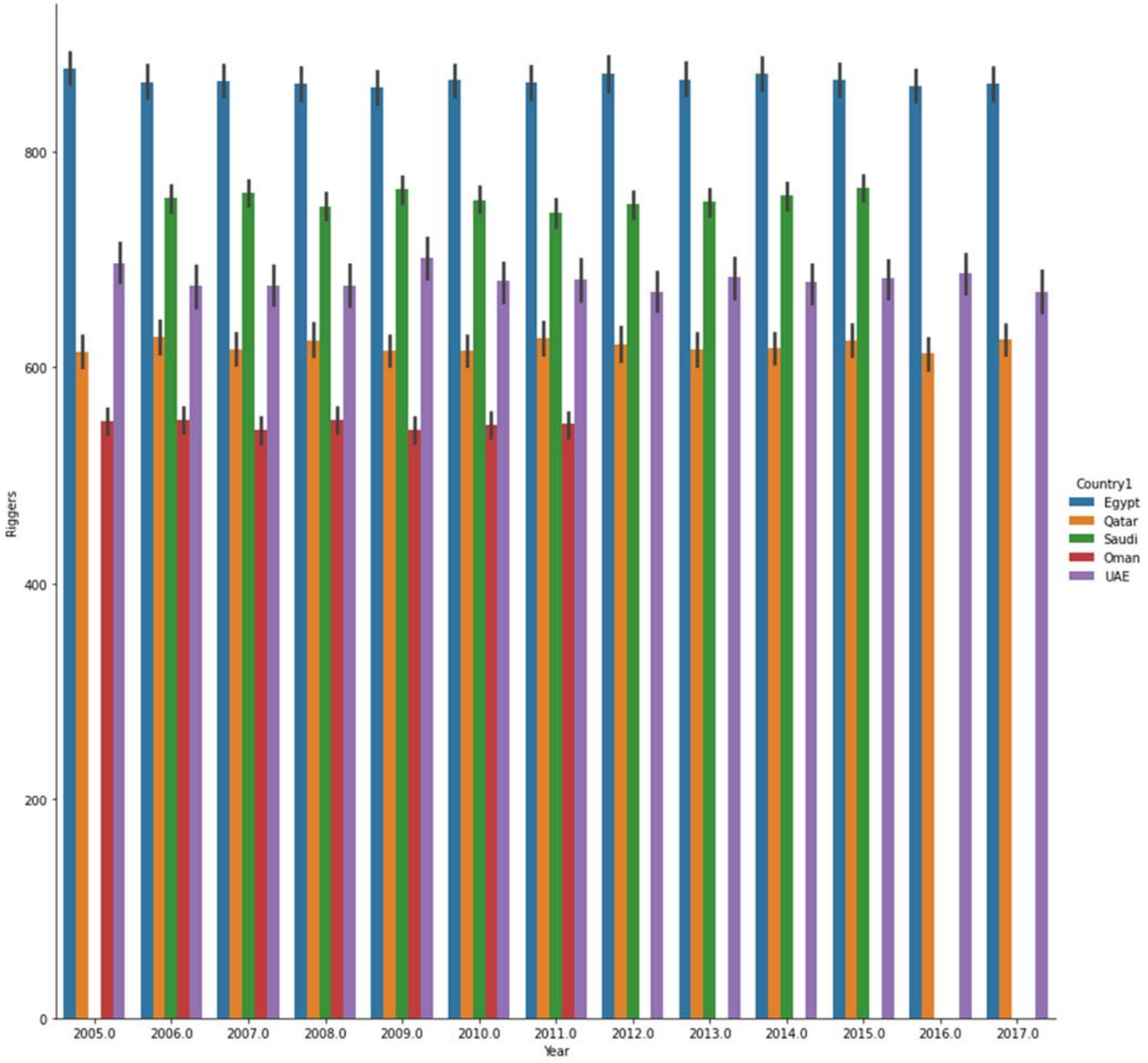
In case the Un-usual circumstances like Heat index is high , availability of material is low , availability of shop drawings is low , political issues is yes , and Working at height is yes have been excluded , we will find that the productivity rate per crew in all countries are increased by average 30%


Average number of Pipe Fitters per Year for Each Country
Pipe Fitter
0
50
100
150
200
250
300
2005.0
2006.0
2007.0
2008.0
2009.0
2010.0
2011.0
2012.0
2013.0
2014.0
2015.0
2016.0
2017.0
Year
Country1
Egypt
Qatar
Saudi
Oman
UAE

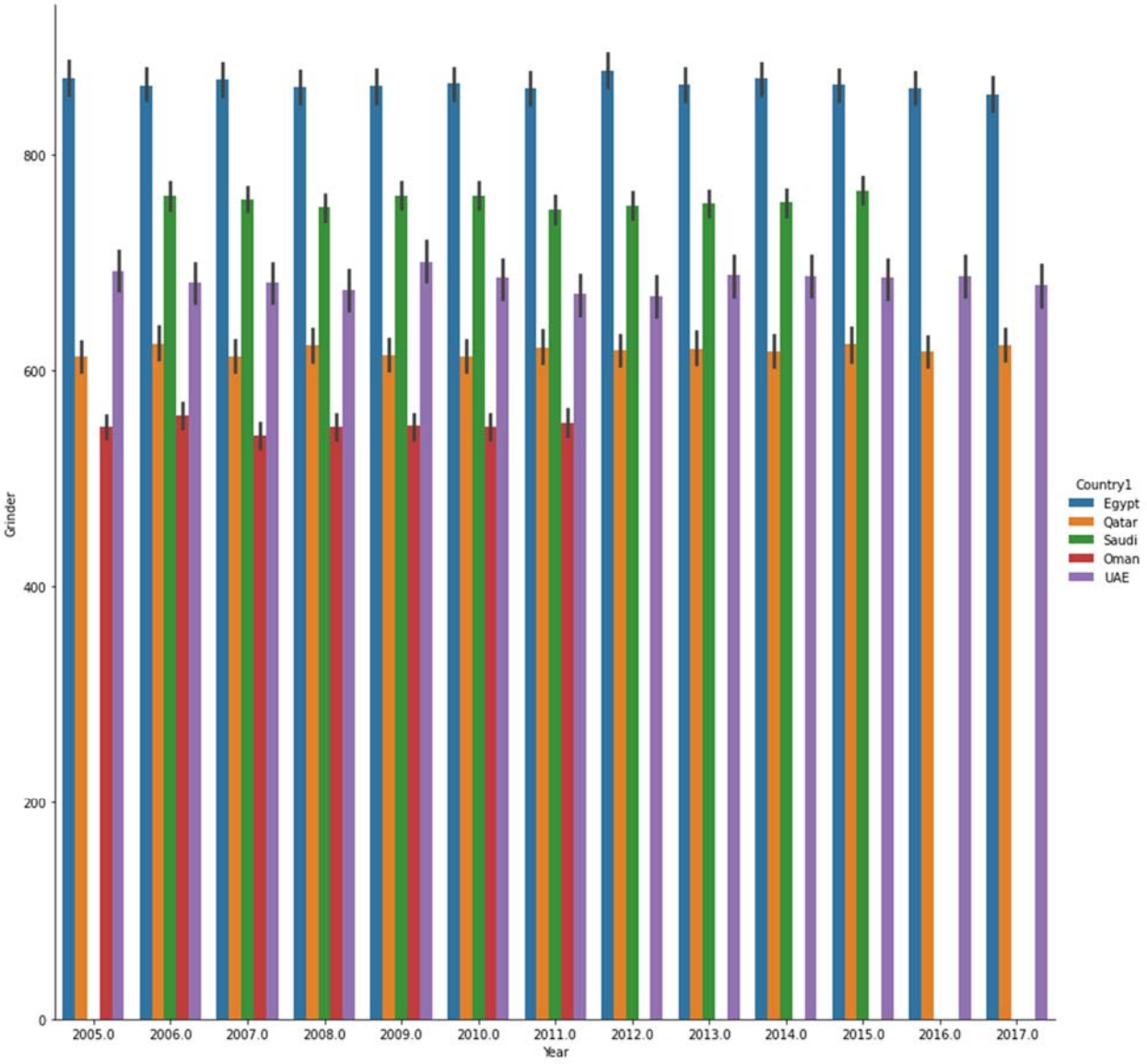
Average number of Cranes per Year for Each Country



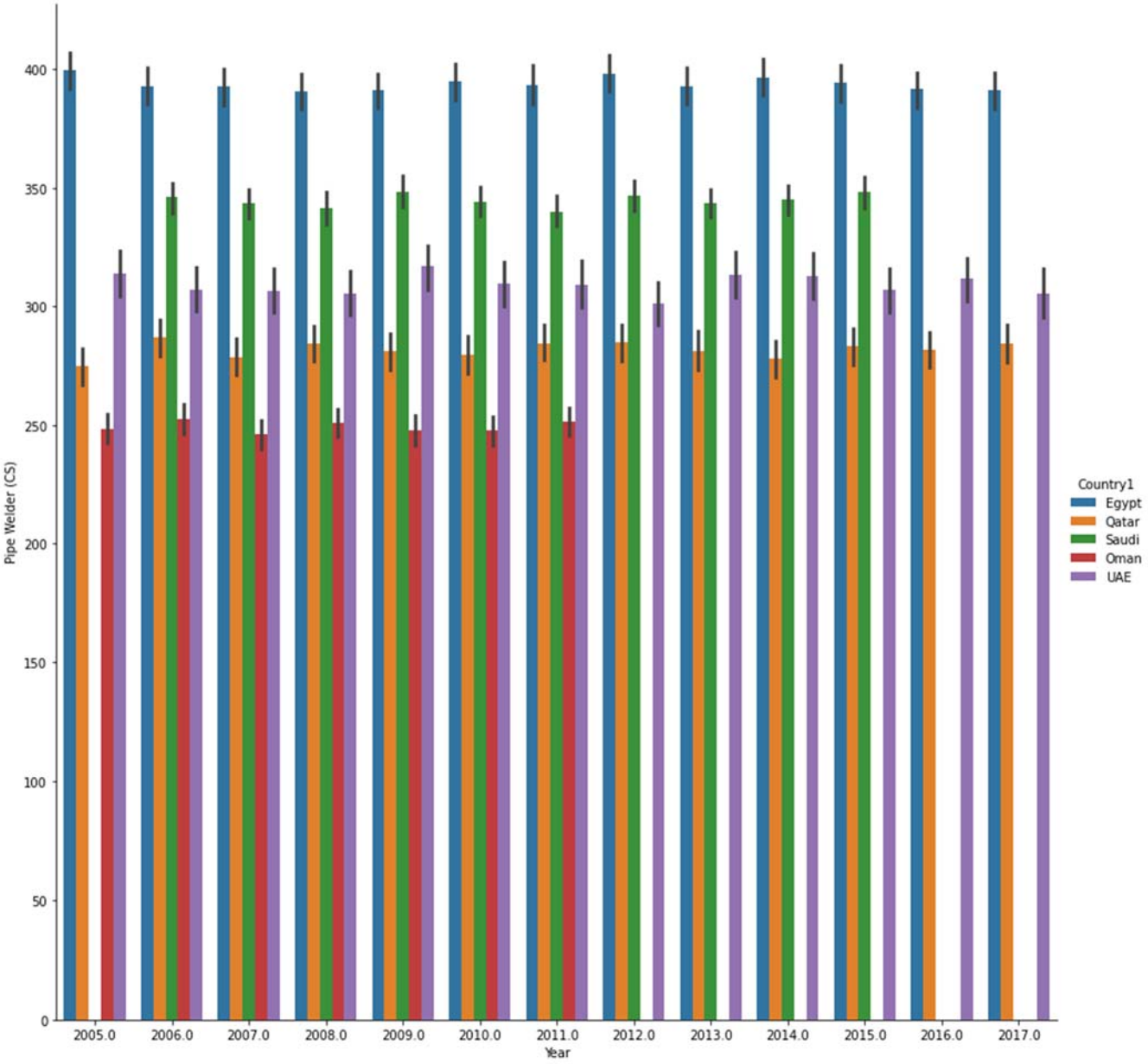
Average number of Riggers per Year for Each Country



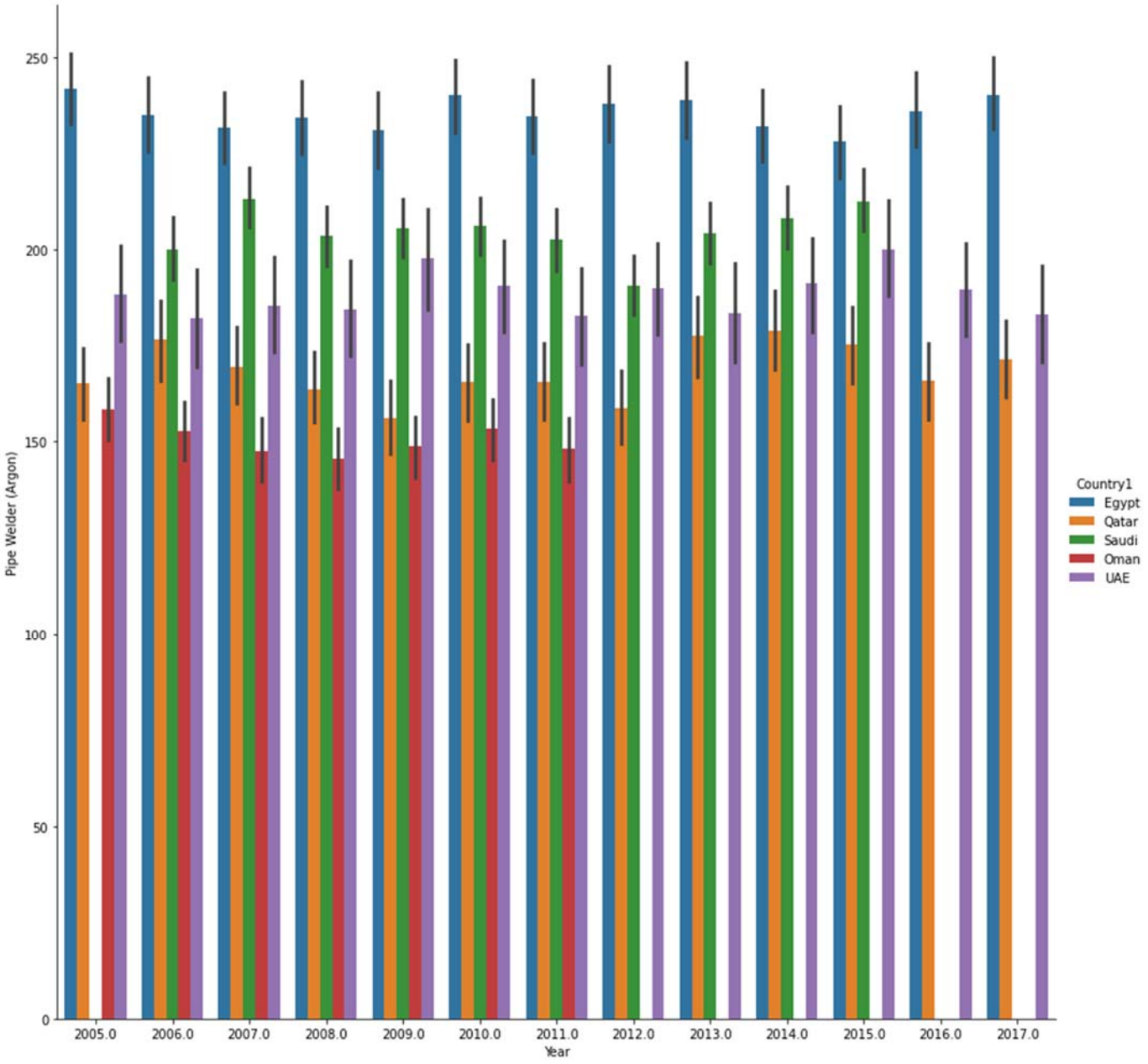
Average number of Grinders per Year for Each Country



Average number of Pipe Welders (C.S) per Year for Each Country

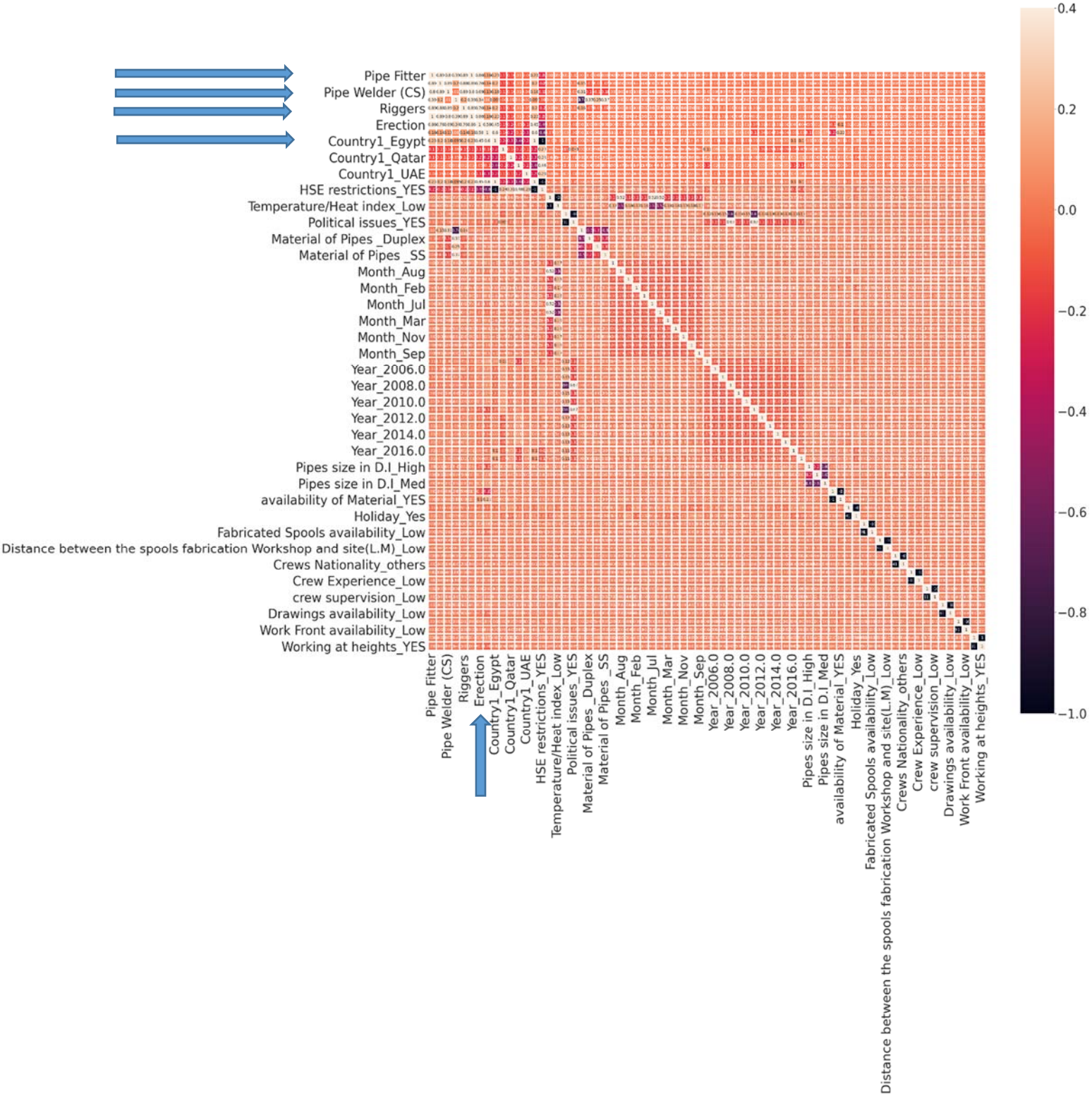


Average number of Pipe Welders (Argon) per Year for Each Country



Heat Map Features

A heatmap is a two-dimensional graphical representation of data where the individual values that are contained in a matrix are represented as colours and reflecting the relation between different features of the Data.



Features which impacting the Erection Production Positively listed in descending order as below:

Pipe Fitter	0.868539
Cranes	0.868539
Grinder	0.773901
Riggers	0.772349
Pipe Welder (CS)	0.699907
Daily_Rate	0.574518
Country1_Egypt	0.450903
HSE restrictions_NO	0.450903
Pipe Welder (Argon)	0.319736
availability of Material_YES	0.097399
Year_2005.0	0.035286
Year_2008.0	0.032485
Pipes size in D.I_Med	0.030312
Political issues_NO	0.028038
Holiday_Yes	0.026085
Working at heights_NO	0.023672
Fabricated Spools availability_High	0.023546
Temperature/Heat index_Low	0.023347
Year_2006.0	0.017812
Year_2009.0	0.016451
Drawings availability_High	0.016303
Year_2007.0	0.016302
Year_2010.0	0.015916
Material of Pipes _CS	0.015312

Obviously, the skilled manpower especially pipe fitters, and equipment like cranes , the location like Egypt , the absence of HSE restrictions , the Low Heat index ,the absence of Holidays , the availability of material and Drawings impacting the Erection Production rate positively.

Features which impacting the Erection Production Negatively listed in ascending order as below:

HSE restrictions_YES	-0.450903
Country1_Qatar	-0.216324
Country1_Oman	-0.189008
Country1_UAE	-0.182628
availability of Material_NO	-0.097399
Year_2011.0	-0.070196
Country1_Saudi	-0.053838
Pipes size in D.I_High	-0.046227
Political issues_YES	-0.028038
Holiday_No	-0.026085
Working at heights_YES	-0.023672
Fabricated Spools availability_Low	-0.023546
Temperature/Heat index_High	-0.023347
Drawings availability_Low	-0.016303
Year_2012.0	-0.014406
Year_2013.0	-0.014027
Month_Aug	-0.013690
Month_Jul	-0.012951
Year_2015.0	-0.012610
Year_2014.0	-0.012255

3. Benchmark Model:

Now, we will train our data in to different models comparing our results with the Benchmark Model.

We will use Linear Regressor model as a Benchmark in which to compare our models' performance to, because it is fast and simple to implement.

We will implement the RMSE (root mean squared error) as a metric to Compare other Models' Results.

4. Algorithms and Techniques:

As we are implementing a Regression Problem, our strategy to implement the models below and comparing their results using our Evaluation metrics to our Benchmark model. Hence, we can assess the best model to be implemented in our Problem.
Admittedly, we will concentrate on the ANN Models like KerasRegressor and Gradient Boosting Models , which Often provides predictive accuracy that cannot be beat , Lots of flexibility - can optimize on different loss functions and provides several hyperparameters tuning options that make the function fit very flexible , No data pre-processing required - often works great with categorical and numerical values as is and Handles missing data .

4.1 Linear Regressor:

Linear regression is probably one of the most important and widely used regression techniques. It's among the simplest regression methods. One of its main advantages is the ease of interpreting results.
When implementing linear regression of some dependent variable y on the set of independent variables $\mathbf{x} = (x_1, \ldots, x_r)$, where r is the number of predictors, you assume a linear relationship between y and $\mathbf{x}$: $y = \beta_0 + \beta_1 x_1 + \cdots + \beta_r x_r + \varepsilon$. This equation is the **regression equation**. $\beta_0, \beta_1, \ldots, \beta_r$ are the **regression coefficients**, and ε is the **random error**.

4.2 Elastic Regressor:

Is a Linear regression with combined L1 and L2 priors as regularize.
a * L1 + b * L2
where :
alpha = a + b and l1_ratio = a / (a + b)

4.3 Ridge Regressor

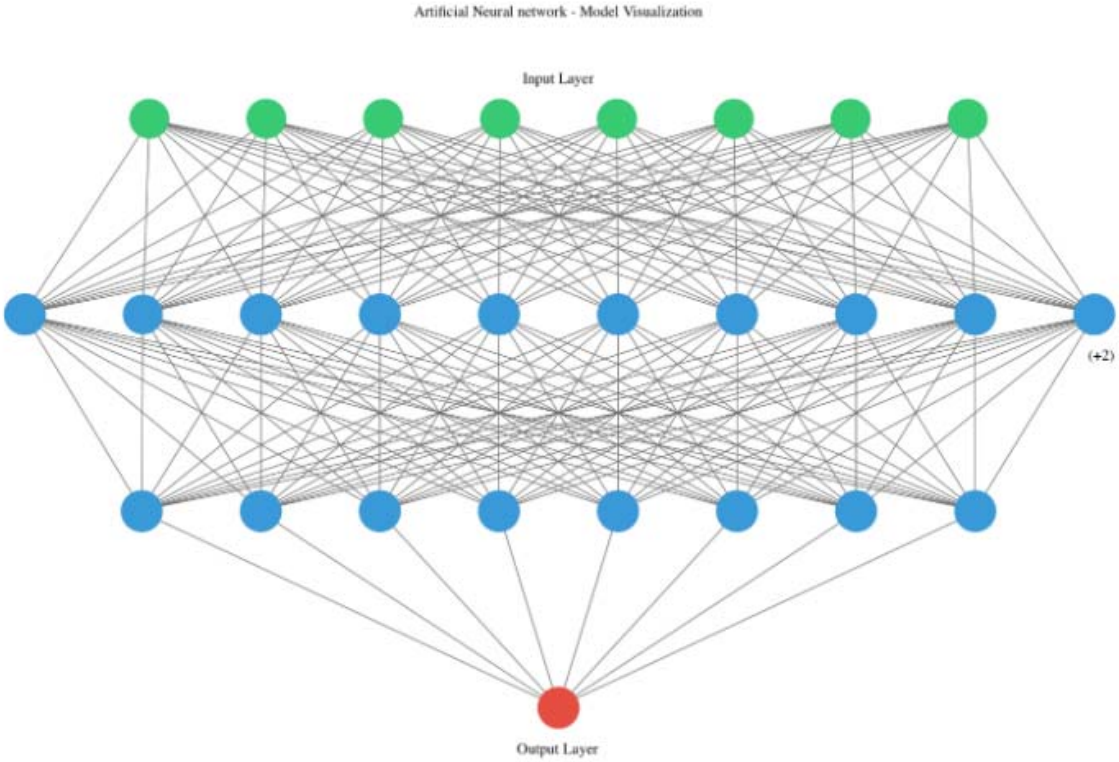
Is a Linear least squares with l2 regularization.
This model solves a regression model where the loss function is the linear least squares function and regularization is given by the l2-norm

4.4 Lasso Regressor

Is a Linear Model trained with L1 prior as regularizer
Technically the Lasso model is optimizing the same objective function as the Elastic Net with l1_ratio=1.0 (no L2 penalty).

4.5 KerasRegressor

The basic architecture of the deep learning neural network, which we will be following, consists of three main components.
1) Input Layer: This is where the training observations are fed. The number of predictor variables is also specified here through the neurons.
2) Hidden Layers: These are the intermediate layers between the input and output layers. The deep neural network learns about the relationships involved in data in this component.
3) Output Layer: This is the layer where the final output is extracted from what's happening in the previous two layers. In case of regression problems, the output later will have one neuron.



4.6 RandomForestRegressor

A random forest is a meta estimator that fits a number of classifying decision trees on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is always the same as the original input sample size.

4.7CatBoostRegressor

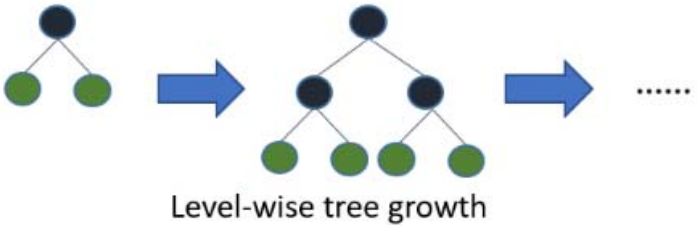
Is a powerful machine learning algorithm that is widely applied to multiple types of business challenges like fraud detection, recommendation items, forecasting and it performs well also. It can also return very good result with relatively less data, unlike DL models that need to learn from a massive amount of data.

4.8 LGBRegressor

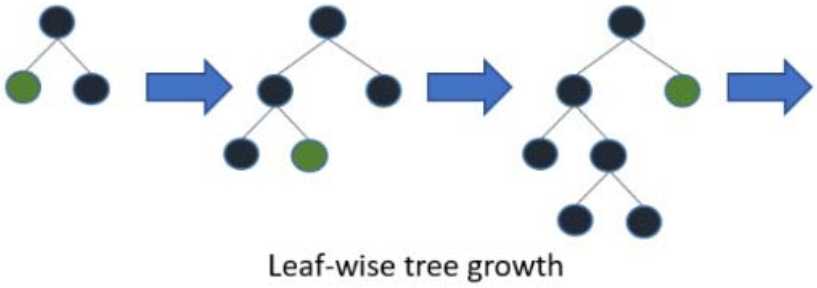
Light GBM is a fast, distributed, high-performance gradient boosting framework based on decision tree algorithm, used for ranking, classification and many other machine learning tasks. Since it is based on decision tree algorithms, it splits the tree leaf wise with the best fit whereas other boosting algorithms split the tree depth wise or level wise rather than leaf-wise. So when growing on the same leaf in Light GBM, the leaf-wise algorithm can reduce more loss than the level-wise algorithm and hence results in much better accuracy which can rarely be achieved by any of the existing boosting algorithms

4.9 XGBRegressor

XGBoost is an implementation of gradient boosted decision trees designed for speed and performance.



Level-wise tree growth in XGBOOST.



Leaf wise tree growth in Light GBM.

3. Model METHODOLOGY

3.1 Data Pre-processing:

- **We will normalize our numerical data for computing speed.**

The below is our numerical data which will be normalized:

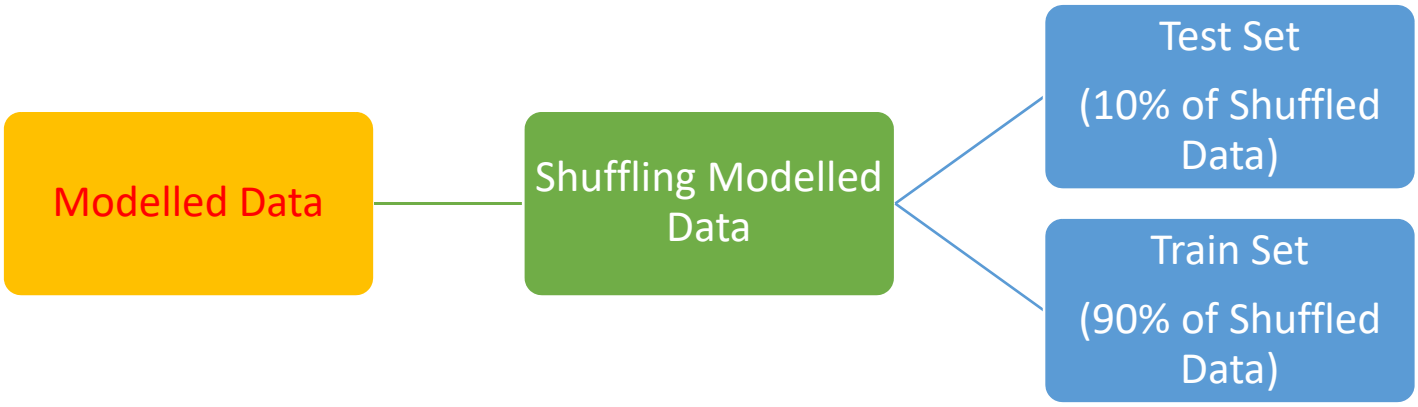
```
from sklearn.preprocessing import MinMaxScaler
mms = MinMaxScaler()
modeled_data[['Pipe Fitter', 'Grinder', 'Pipe Welder (CS)', 'Pipe Welder (Argon)','Daily_Rate',
       'Riggers', 'Cranes']]= mms.fit_transform(modeled_data[['Pipe Fitter', 'Grinder', 'Pipe Welder (CS)', 'Pipe Welder (Argon)','Daily_Rate',
       'Riggers', 'Cranes']])
```

- **We will implement dummies(0's and 1's) for our Categorical data**

The below is our dummies List:

```
dummies_list=['Country1', 'HSE restrictions','Temperature/Heat index', 'Political issues', 'Material of Pipes ','Month','Year','Pipes size in D.I',
       'availability of Material', 'Holiday','Fabricated Spools availability','Distance between the spools fabrication Workshop and site(L.M)',
       'Crews Nationality', 'Crew Experience', 'crew supervision','Drawings availability', 'Work Front availability','Working at heights']


for col in dummies_list:
    dummy = pd.get_dummies(modeled_data[col]).rename(columns=lambda x: col+'_' +str(x))
    modeled_data = pd.concat([modeled_data, dummy], axis = 1)
modeled_data.drop(dummies_list, axis=1, inplace = True)
```

- **Preparation of our Data for Models Training and testing**



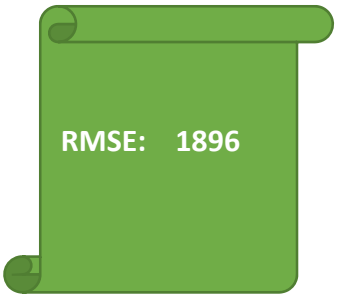
We will shuffle our data, then it will be divided by 90 % for train data an 10 % for test data

3.2 Implementation:

Firstly - after the Preparation of our training and testing data sets -We Will implement our Benchmark model (Linear regression Model) and calculating our Metrics that we have discussed before.

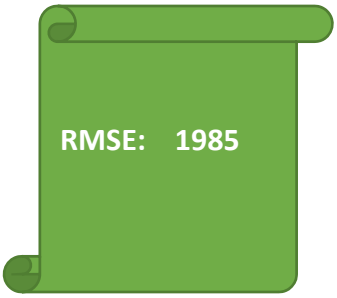
3.2.1 Linear REGRESSION MODEL (BENCHMARK MODEL):

```
from sklearn.model_selection import GridSearchCV
from sklearn.linear_model import ElasticNet
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error


regression=LinearRegression()
regression.fit(X_train.values, y_train.values)
y_pred_linear_regression=regression.predict(X_dev.values)
y_Actual=y_dev
Linear_RMSE=sqrt(mean_squared_error(y_Actual,y_pred_linear_regression))
print('Linear Regression_RMSE:',Linear_RMSE)
```



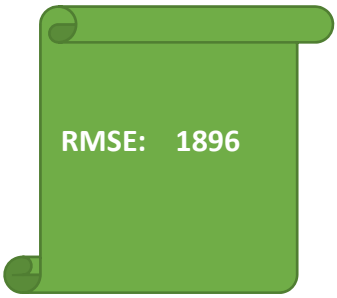
3.2.2 Elastic Regressor:

```
elastic=ElasticNet(normalize=True,alpha=1e-05,l1_ratio=0.8)
elastic.fit(X_train.values, y_train.values)
y_pred_linear_elastic=elastic.predict(X_dev.values)
y_Actual=y_dev
elastic_model_RMSE=sqrt(mean_squared_error(y_Actual,y_pred_linear_elastic))
print('Elastic Regression_RMSE:',elastic_model_RMSE)
```



3.2.3 Ridge Regressor:

```
from sklearn.linear_model import Ridge
ridgereg = Ridge(alpha=0.0001,normalize=True)
ridgereg.fit(X_train.values, y_train.values)
y_pred_ridge = ridgereg.predict(X_dev.values)
y_Actual=y_dev
Ridge_model_RMSE=sqrt(mean_squared_error(y_Actual,y_pred_ridge))
print('Ridge Regression_RMSE:',Ridge_model_RMSE)
```

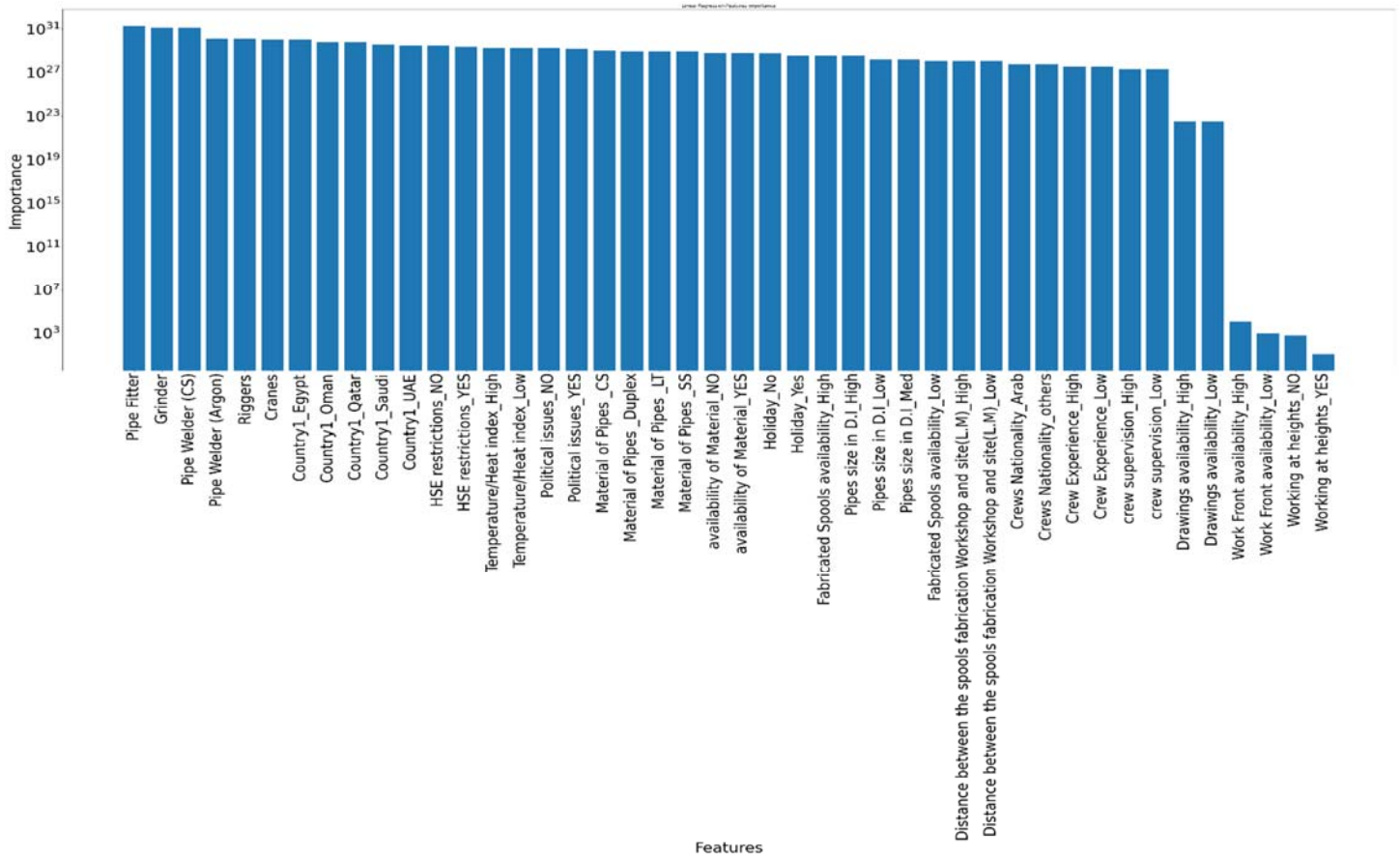


3.2.4 Lasso Regressor:

```
from sklearn.linear_model import Lasso
lassoreg = Lasso(alpha=1)
lassoreg.fit(X_train.values, y_train.values)
y_pred_lasso = lassoreg.predict(X_dev.values)
y_Actual=y_dev
lassoreg_model_RMSE=sqrt(mean_squared_error(y_Actual,y_pred_lasso))
print('Lasso Regression_RMSE:',lassoreg_model_RMSE)
```

RMSE: 1896

Features importance for the Linear, Ridge, Lasso and Elastic Regressors:



Obviously, the highest features importance are the skilled labours like, Pipe fitters, Grinders and Welders

3.2.5 Keras Regressor:

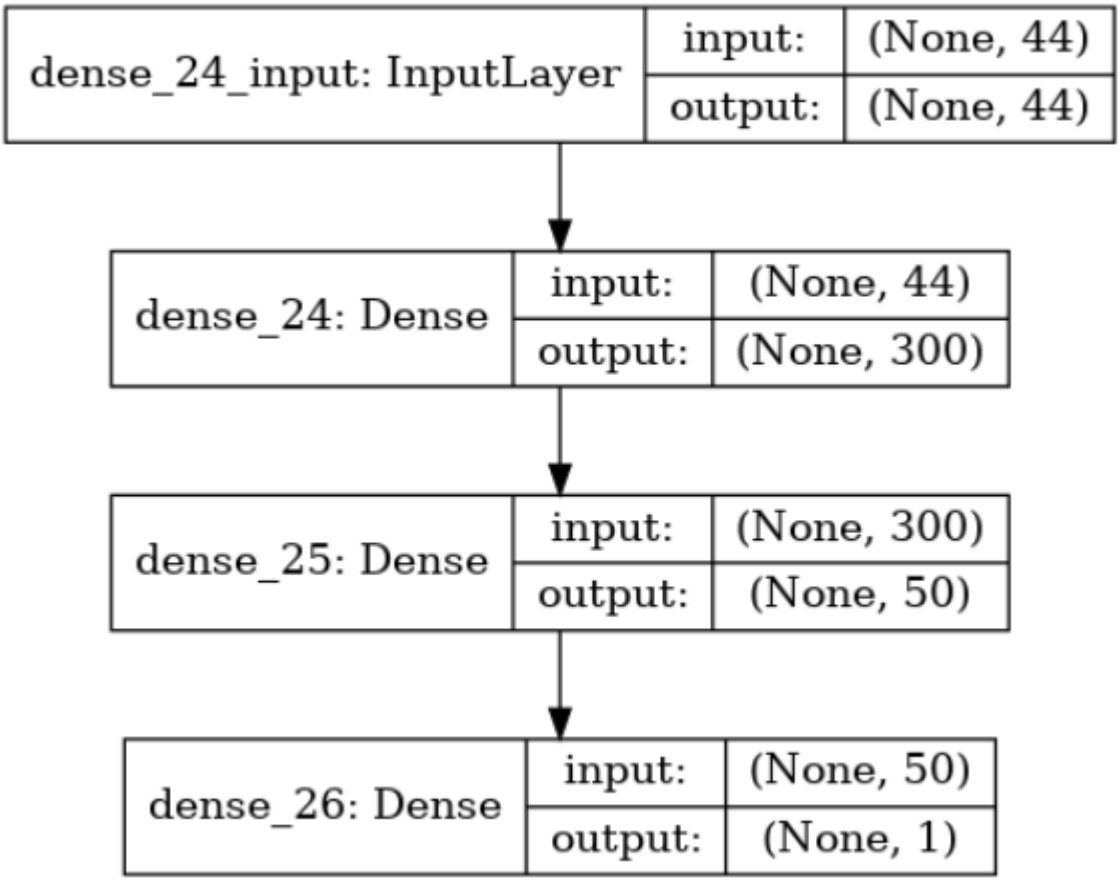
RMSE: 1330

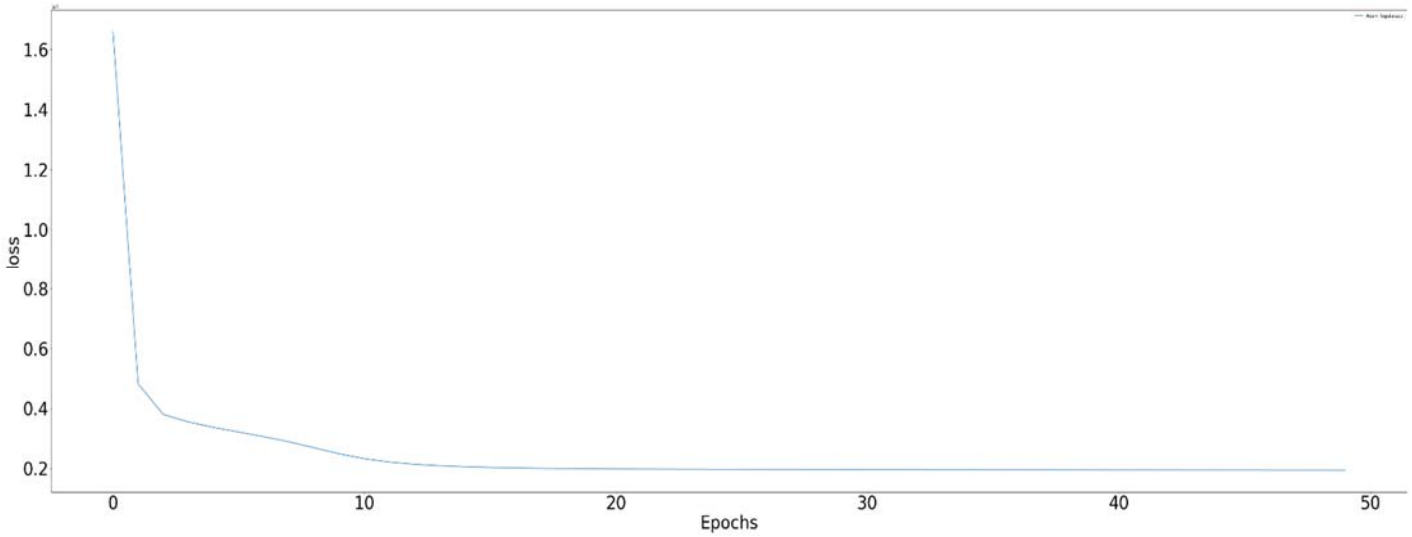
```
from keras.optimizers import Adam
from keras.callbacks import EarlyStopping
opt = Adam(lr=1e-3, decay=1e-3 / 200)
def base_model():
    alpha=.00001
    model = Sequential()
    model.add(Dense(300, input_dim=X_train.shape[1],kernel_regularizer = regularizers.l2(0.01), kernel_initializer='normal' ,activation='relu'))
    model.add(Dense(50,kernel_regularizer = regularizers.l2(0.01), kernel_initializer='normal',activation='relu',))
    model.add(Dense(1, kernel_initializer='normal',activation='linear'))
    model.compile(loss='mean_squared_error', optimizer= opt)
    return model

reg = KerasRegressor(build_fn=base_model, epochs=50, batch_size=100,verbose=1,validation_split=0.1)
kfold = KFold(n_splits=5, random_state=43)
results = np.sqrt(-1*cross_val_score(reg, X_train.values, y_train.values,scoring= "neg_mean_squared_error", cv=kfold))
print("Training RMSE mean and std from CV: {} {}".format(results.mean(),results.std()))
```

- **Model Summary:**

```
reg.model.summary()
```

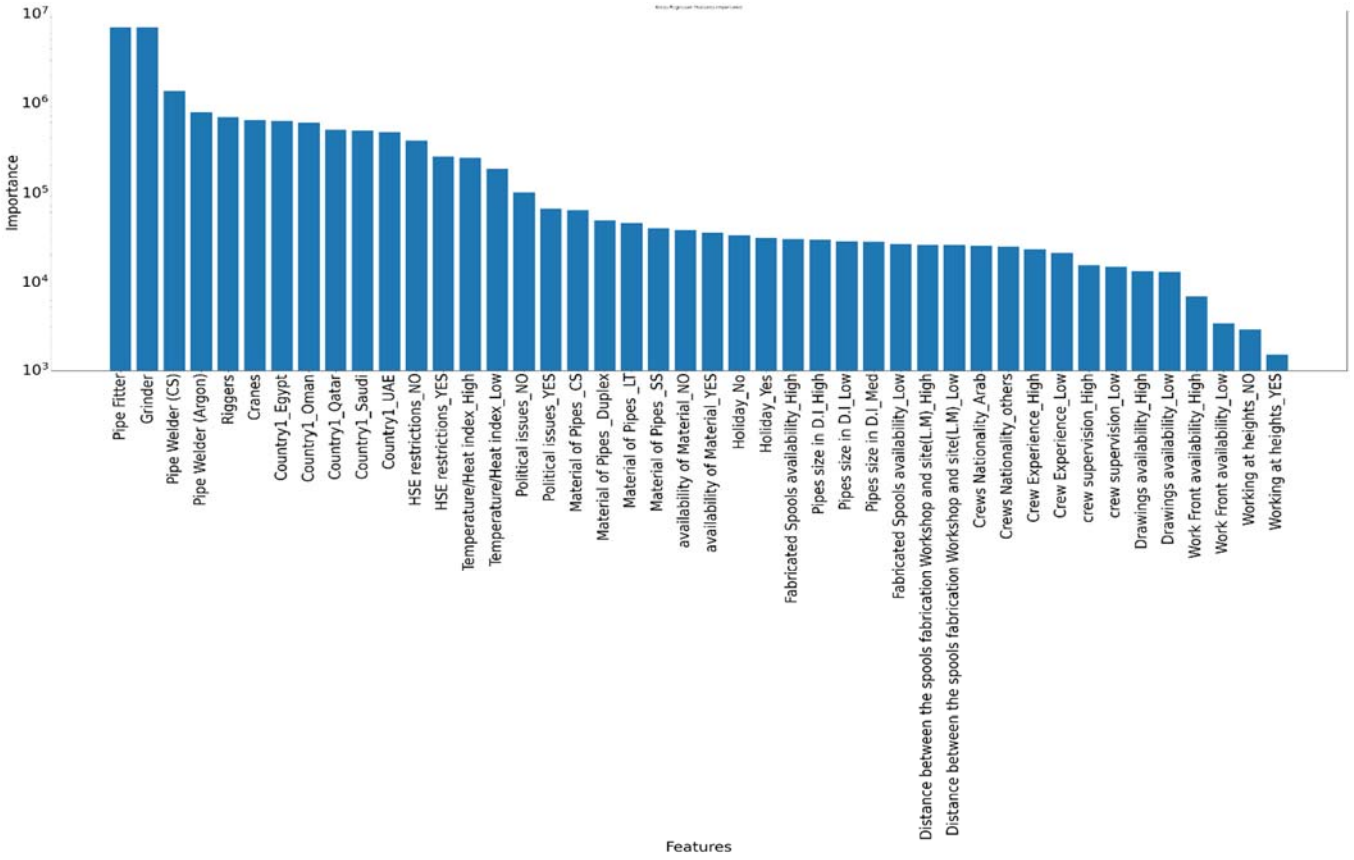
```
Model: "sequential_6"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_16 (Dense)             (None, 300)               13500
_________________________________________________________________
dense_17 (Dense)             (None, 50)                15050
_________________________________________________________________
dense_18 (Dense)             (None, 1)                 51
=================================================================
Total params: 28,601
Trainable params: 28,601
Non-trainable params: 0
_________________________________________________________________
```





As seen, by increasing the number of Epochs, the loss is decreased, till reach the optimum at 50 epochs

Features importance for Keras Regressor:



Obviously, the highest features importance are the skilled labours like, Pipe fitters, Grinders and Welders

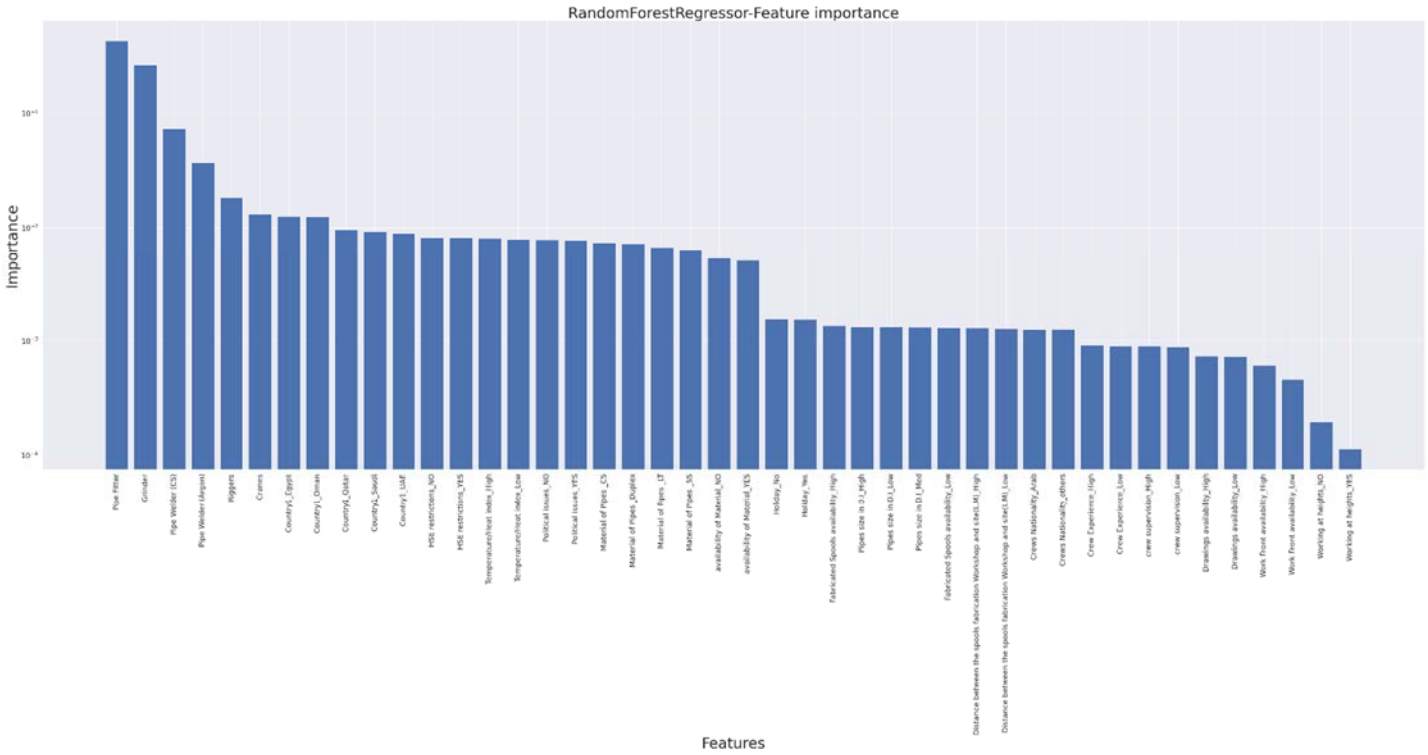
3.2.6 RandomForestRegressor:

```
#Random forest model specification
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import confusion_matrix
def binary(movement):
    """
    Converts percent change to a binary 1 or 0, where 1 is an increase and 0 is a decrease/no change

    """
    #Empty arrays where a 1 represents an increase in price and a 0 represents a decrease in price
    direction = np.empty(movement.shape[0])
    #If the change in price is greater than zero, store it as a 1
    #If the change in price is less than zero, store it as a 0
    for i in range(movement.shape[0]):
        if movement[i] > 0:
            direction[i] = 1
        else:
            direction[i]= 0
    return direction


regr = RandomForestRegressor(n_estimators=20, criterion='mse', max_depth=None,
                     min_samples_split=2, min_samples_leaf=1,
                     min_weight_fraction_leaf=0.0, max_features='auto',
                     max_leaf_nodes=None, min_impurity_decrease=0.0,
                     min_impurity_split=None, bootstrap=True,
                     oob_score=False, n_jobs=1, random_state=None,
                     verbose=2, warm_start=False)

#Train on data
regr.fit(X_train, y_train.ravel())

y_pred_random = regr.predict(X_dev)
y_dev = y_dev.to_frame()
```

RMSE: 1429

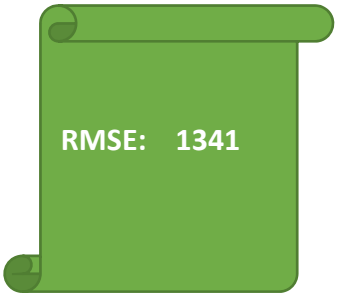
```
y_pred_random = regr.predict(X_dev)
RandomForrest_rmse = sqrt(mean_squared_error(y_dev, y_pred_random))
print('RandomForestRegressor_RMSE:',RandomForrest_rmse)
```

Features importance for RandomForestRegressor:



3.2.7 CatBoostRegresso :

```
from catboost import CatBoostRegressor


#model_cat = CatBoostRegressor(iterations=5000, learning_rate=0.05, depth=5)
# Fit model
#reg_cv.fit(X_train,y_train,eval_metric='rmse', verbose = True, eval_set = [(X_dev, y_dev)])


# Get predictions
#preds = model.predict(test_pool)
model_cat = CatBoostRegressor()
parameters = {'depth'         : [6,8,10],
                  'learning_rate' : [0.01, 0.05, 0.1],
                  'iterations'    : [30, 50, 100]
                 }
grid = GridSearchCV(estimator=model_cat, param_grid = parameters, cv = 2, n_jobs=-1)
grid.fit(X_train, y_train)

    # Results from Grid Search
print("\n========================================================")
print(" Results from Grid Search " )
print("========================================================")

print("\n The best estimator across ALL searched params:\n",
          grid.best_estimator_)

print("\n The best score across ALL searched params:\n",
          grid.best_score_)

print("\n The best parameters across ALL searched params:\n",
          grid.best_params_)

print("\n ========================================================")
```




```
CatBoostRegressor_RMSE = sqrt(mean_squared_error(y_dev, y_pred_catboost))
print('CatBoostRegressor_RMSE:',CatBoostRegressor_RMSE)
```

3.2.7 LGBRegressor :

```
import lightgbm as lgb
'''lgb_model = lgb.LGBMRegressor(max_depth=6,
                                 colsample_bytree=0.8,
                                 learning_rate=0.1,
                                 n_estimators=500,
                                 subsample=0.8
                             )'''
lgb_model = lgb.LGBMRegressor()
parameters = {'max_depth'         : [6,8,10],
                  'learning_rate' : [0.01, 0.05, 0.1],
                  'iterations'    : [30, 50, 100,500]
                 }
grid = GridSearchCV(estimator=lgb_model, param_grid = parameters, cv = 2, n_jobs=-1)
grid.fit(X_train, y_train)

    # Results from Grid Search
print("\n========================================================")
print(" Results from Grid Search " )
print("========================================================")

print("\n The best estimator across ALL searched params:\n",
          grid.best_estimator_)

print("\n The best score across ALL searched params:\n",
          grid.best_score_)

print("\n The best parameters across ALL searched params:\n",
          grid.best_params_)

print("\n ========================================================")
```


```
LGBMRegressor_RMSE = sqrt(mean_squared_error(y_dev, y_pred_lgb))
print('LGBMRegressor_RMSE:',LGBMRegressor_RMSE)
```

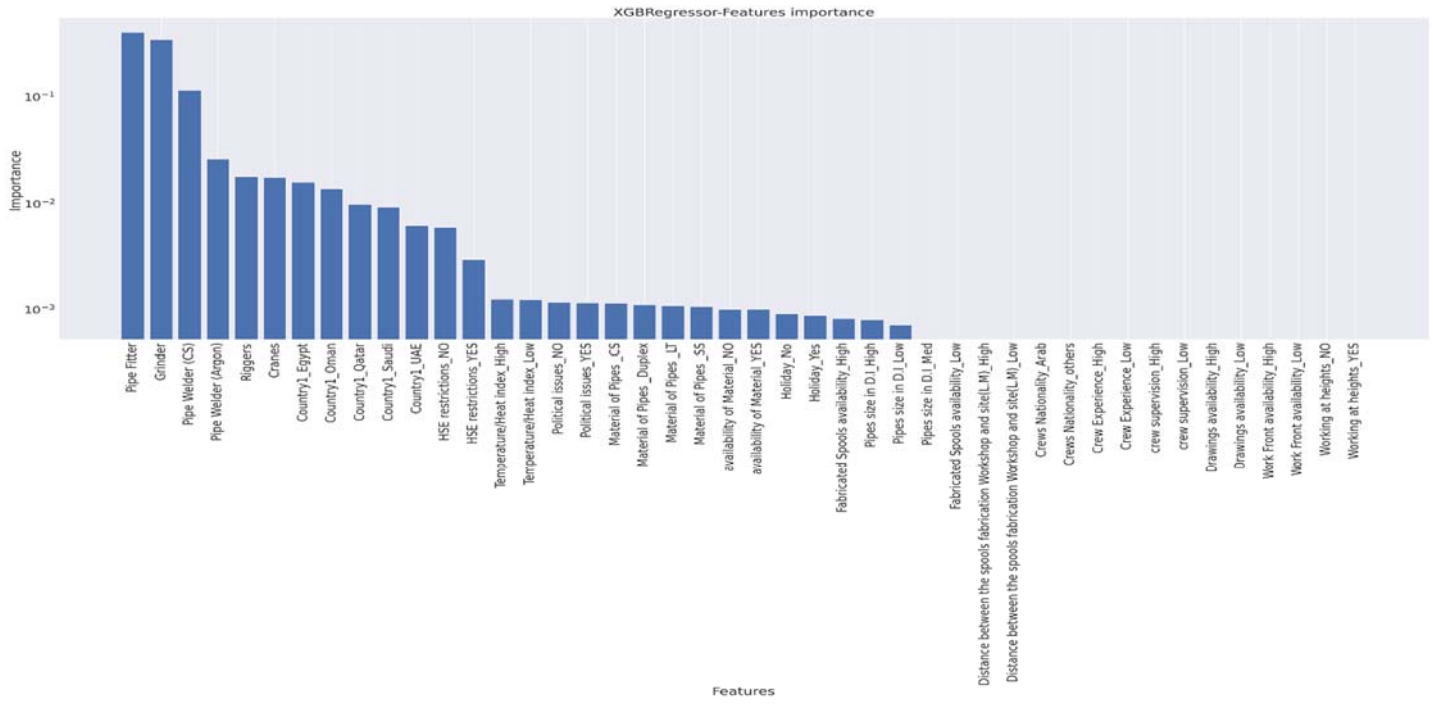
3.2.7 XGBRegressor :

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from sklearn.model_selection import StratifiedKFold
import xgboost as xgb

from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import accuracy_score
from sklearn.model_selection import RandomizedSearchCV,GridSearchCV


gbm = xgb.XGBRegressor(objective="reg:linear", seed=1729)
reg_cv = GridSearchCV(gbm, {"colsample_bytree":[1.0],"min_child_weight":[1.2]
                            ,'max_depth': [5], 'n_estimators': [100]}, verbose=1)
reg_cv.fit(X_train,y_train,eval_metric='rmse', verbose = True, eval_set = [(X_dev, y_dev)])
```

RMSE: 1421

```
XGBRegressor_RMSE = sqrt(mean_squared_error(y_dev, y_pred_xgboost))
print('XGBRegressor_RMSE:',XGBRegressor_RMSE)
```

Features importance for XGB,LGB and Cat regressors:



Obviously, the highest features importance are the skilled labours like, Pipe fitters, Grinders and Welders

4. Results:

4.1 Models Evaluation:

Firstly, we will Combine all the models together and sort their Results according to our Metric (RMSE)

	Model_Type	RMSE
0	KerasRegressor	1330.432233
1	CatBoostRegressor	1341.735370
2	LGBRegressor	1342.716727
3	XGBRegressor	1421.338866
4	RandomForestRegressor	1429.713910
5	Lasso_regression	1896.831556
6	Ridge_regression	1896.936406
7	Linear_regression	1896.965860
8	Elastic_regression	1985.370321

Obviously, The Keras regressor is the best Model with least RMSE (1330), and this RMSE is acceptable as we are working with target in thousands, then the Boosting Models (Cat, LGB and XGB) come in the second rank with RMSE (1341) which are so close to the Keras regressor.

Secondly, we will test our Model with new data and we will observe the output:

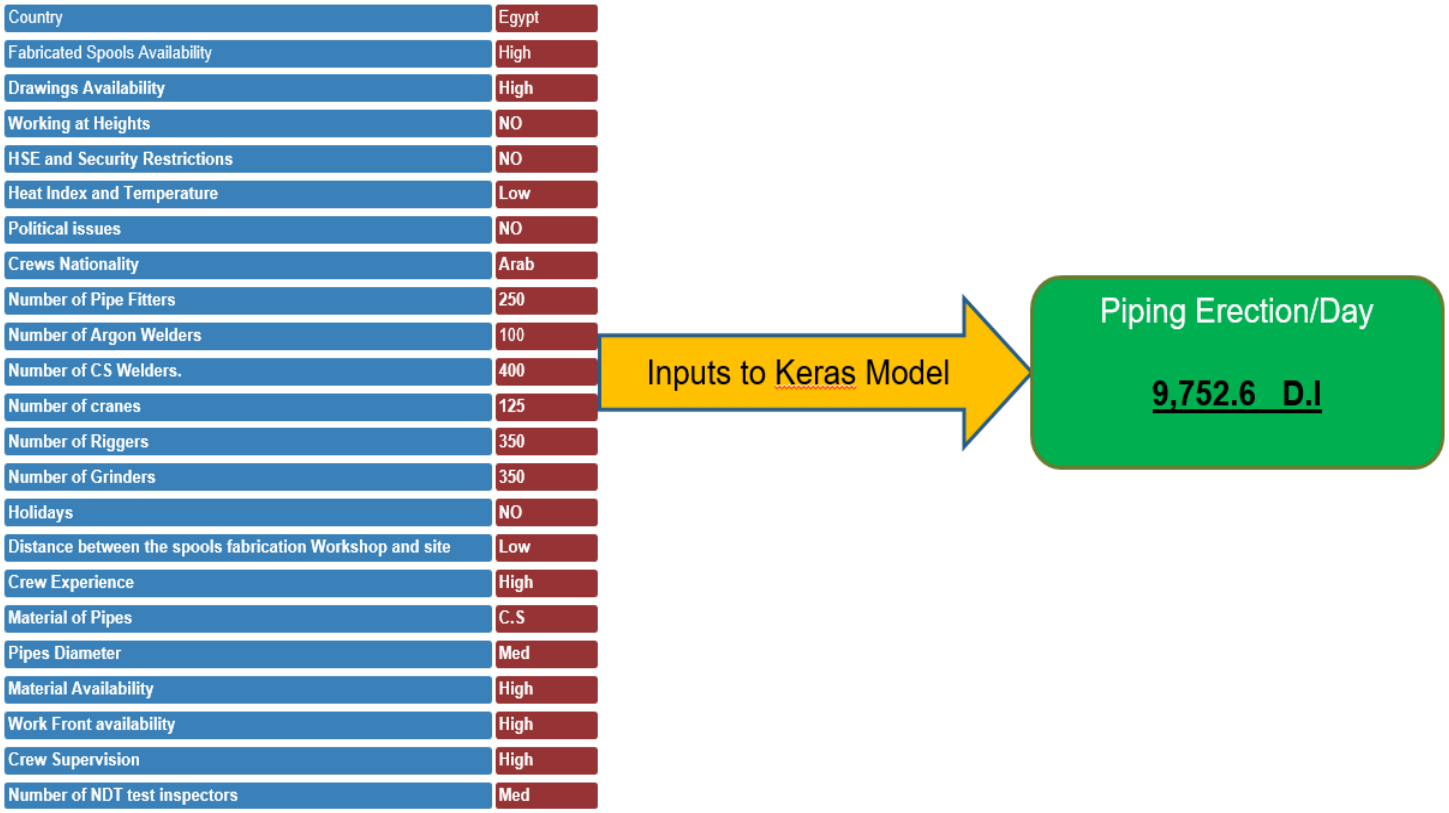
The first selection it will be as below:

We will select Country “Egypt”, with the features as below

```
from keras.models import load_model

# Instantiate the model as you please (we are not going to use this)
model2 = KerasRegressor(build_fn=base_model, epochs=50, batch_size=100, verbose=1)

# This is where you load the actual saved model into new variable.
model2.model = load_model('/kaggle/input/kerasmodel/saved_model.h5')

# Now you can use this to predict on new data (without fitting model2, because it uses the older saved model)
model2.predict(X_new)
```

```
1/1 [==============================] - 0s 21ms/step
```

```
array([9752.665], dtype=float32)
```



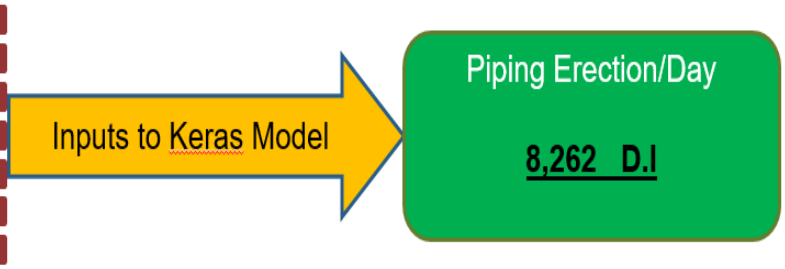
The Second Selection will change the Country to (Qatar) , With HSE restriction (YES) ,and we will fix other features as below:

```
from keras.models import load_model

# Instantiate the model as you please (we are not going to use this)
model2 = KerasRegressor(build_fn=base_model, epochs=50, batch_size=100, verbose=1)

# This is where you load the actual saved model into new variable.
model2.model = load_model('/kaggle/input/kerasmodel/saved_model.h5')

# Now you can use this to predict on new data (without fitting model2, because it uses the older saved model)
model2.predict(X_new)
```

```
1/1 [==============================] - 0s 9ms/step
```

```
array([8262.763], dtype=float32)
```

Country	Qatar
Fabricated Spools Availability	High
Drawings Availability	High
Working at Heights	NO
HSE and Security Restrictions	YES
Heat Index and Temperature	Low
Political issues	NO
Crews Nationality	Others
Number of Pipe Fitters	250
Number of Argon Welders	100
Number of CS Welders.	400
Number of cranes	125
Number of Riggers	350
Number of Grinders	350
Holidays	NO
Distance between the spools fabrication Workshop and site	Low
Crew Experience	High
Material of Pipes	C.S
Pipes Diameter	Med
Material Availability	High
Work Front availability	High
Crew Supervision	High
Number of NDT test inspectors	Med

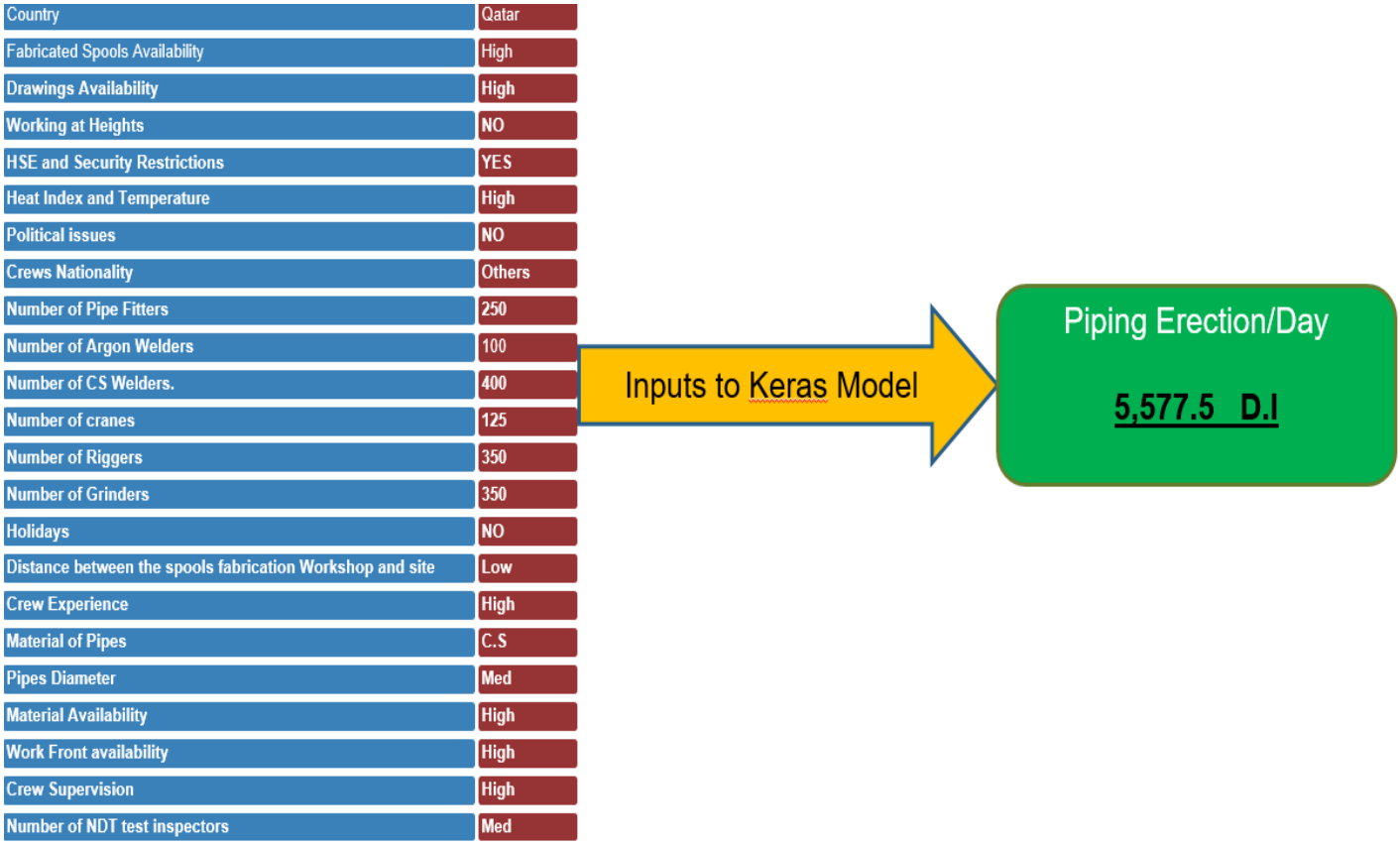


<u>The Third Selection</u> will keep the same features as per second selection except, that we will change the <u>Heat index to high</u>

```
from keras.models import load_model

# Instantiate the model as you please (we are not going to use this)
model2 = KerasRegressor(build_fn=base_model, epochs=50, batch_size=100, verbose=1)

# This is where you load the actual saved model into new variable.
model2.model = load_model('/kaggle/input/kerasmodel/saved_model.h5')

# Now you can use this to predict on new data (without fitting model2, because it uses the older saved model)
model2.predict(X_new)
```

```
1/1 [==============================] - 0s 9ms/step
```

```
array([5577.5674], dtype=float32)
```

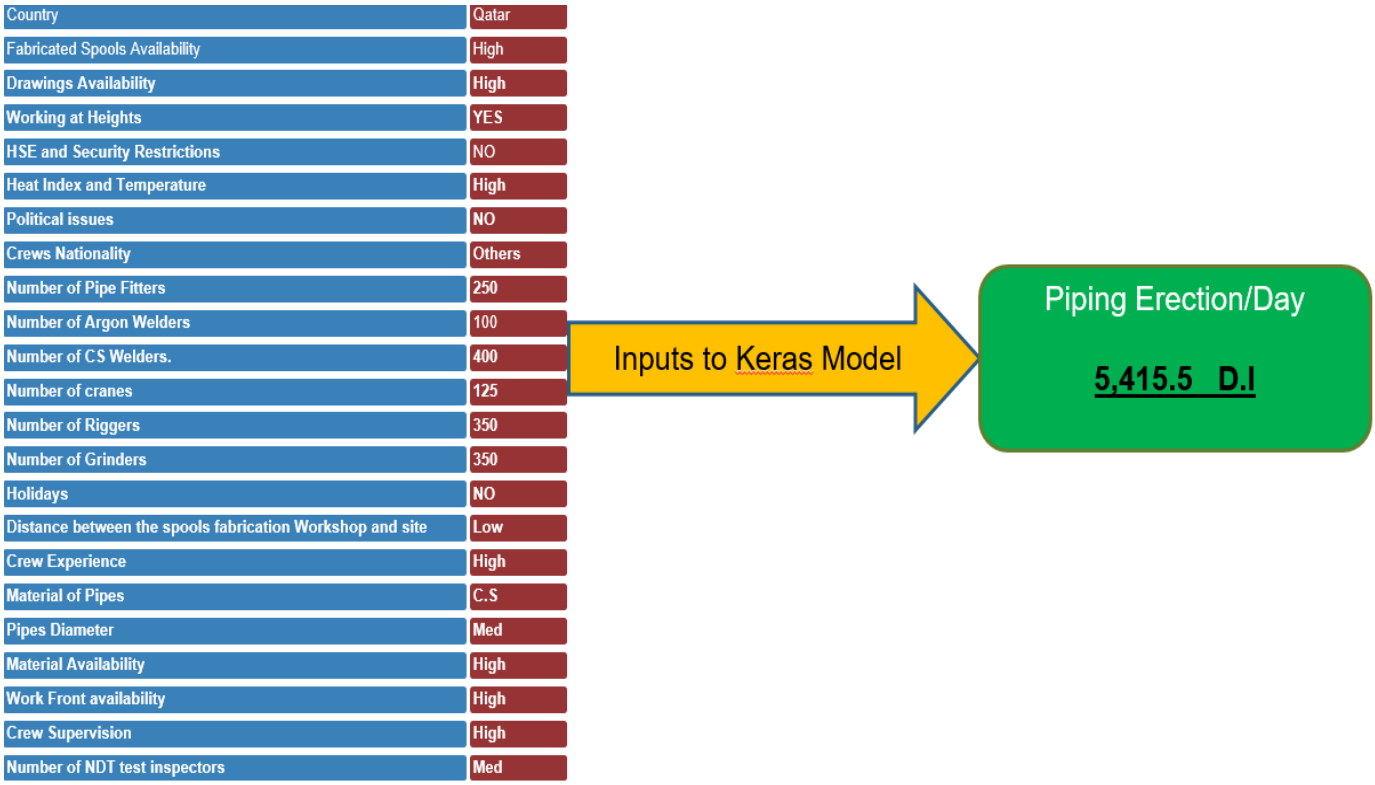


Obviously, the drop in production due to the Heat index is high in summer season.

<u>The Fourth Selection,</u> we will keep the same features as per third Selection except we will change the <u>Working at height to (YES)</u>

```
from keras.models import load_model

# Instantiate the model as you please (we are not going to use this)
model2 = KerasRegressor(build_fn=base_model, epochs=50, batch_size=100, verbose=1)

# This is where you load the actual saved model into new variable.
model2.model = load_model('/kaggle/input/kerasmodel/saved_model.h5')

# Now you can use this to predict on new data (without fitting model2, because it uses the older saved model)
model2.predict(X_new)
```

```
array([5415.35921576])
```



Obviously, there is <u>more drop</u> in piping production in case we are working at heights.

4.2 Justification and Conclusion:

- The Skilled Labours are the most important features according to the most of the models, then the project location, HSE restriction, Temperature /Heat index, political issues, Type of Material,… consecutively as shown in the features importance.
- Artificial intelligence and Machine Learning considered to be the Future Success Key for any Company. The Pioneers in the Future are those who take into account the Artificial intelligence and Machine Learning in their decisions, and in developing management techniques.
- The Historical Data is the cornerstone to build a good model with good results, Companies has to collect and collect and collect data and records , everything to be collected : daily reports , Area of concerns , manpower reports , equipment reports , accidents , incidents ,Daily temperatures,….etc.
- The research can be applied for different trades like Shuttering, steel fixing, piping fabrication, equipment installation, concrete pouring, steel erection,…etc. , and then we can implement standard rate for all trades .

5. References :

- https://towardsdatascience.com/handling-imbalanced-datasets-in-machine-learning-7a0e84220f28
- https://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_auc_score.html
- https://aws.amazon.com/blogs/machine-learning/simplify-machine-learning-with-xgboost-and-amazon-sagemaker/
- http://uc-r.github.io/gbm_regression
- https://www.analyticsvidhya.com/blog/2017/08/catboost-automated-categorical-data/
- https://en.wikipedia.org/wiki/Receiver_operating_characteristic
- https://www.kaggle.com/arthurtok/introduction-to-ensembling-stacking-in-python
- https://stackoverflow.com/questions/10373660/converting-a-pandas-groupby-object-to-dataframe

Perfection is not attainable, but if we chase perfection we can catch excellence.

Vince Lombardi