

# **FINAL PROJECT REPORT**

## **Field Profitability Index Prediction**

Submitted by:

Andi Pratama

Wilson Robert Pariangan

Andrew Wisnujati

**Project Lab Data Science  
Data Science Non Degree Program  
Pacmann Academy**

# Table of Content

<b>Table of Content</b>	<b>2</b>
<b>Background</b>	<b>3</b>
<b>Objectives</b>	<b>3</b>
<b>Process Overview</b>	<b>3</b>
3.1. Dataset & Features	4
3.2. Data Cleaning	5
3.3. Feature Engineering	9
3.4. Exploratory Data Analysis	13
3.5. Modeling Process	20
<b>Product</b>	<b>22</b>
4.1. User Persona	22
4.2. Product Features	22
4.3. Result	24
<b>Conclusion</b>	<b>25</b>
<b>References</b>	<b>25</b>
<b>Github Links</b>	<b>25</b>
<b>App Links</b>	<b>25</b>

## 1. Background

When oil companies discover a field, they have to make an early economic calculation to determine the future of the discovered field, either to be developed or abandoned. Thus, they need a model that can calculate the preliminary economic (economic scale) in a short time before handing over the field to the development team to gain significant information before designing the development concept.

Indonesia has 100 years of oil and gas field data and with the fast computing engine and big data analytics algorithm, we can utilize the data to make predictions on the future commercial of the field.

## 2. Objectives

In this project we used a dataset from Indonesian Oil and Gas Government Institution, consisting of subsurface features such as Field Name, Oil and Gas Inplace, and other subsurface parameters including economic indicator for each field.

Our goal is to provide a benchmarking tool for Oil & Gas participants to get an early indicative economic value such as Profitability Index (PI) of newly discovered oil and gas fields. This information is crucial to enhance the field development concept from a technical and economic point of view because engineer/ development staf/ entity could focus more on the fields that have higher chance to become profitable so they are able to increase their productivity and accelerate development stage.

## 3. Process Overview

In this project we try to define whether an oil and gas field is profitable to be further developed or not together with its probability. We also used several models such as Decision Tree, XGBoost, LGBM and Deep Learning. The flow chart below illustrates the process we are doing in this project.

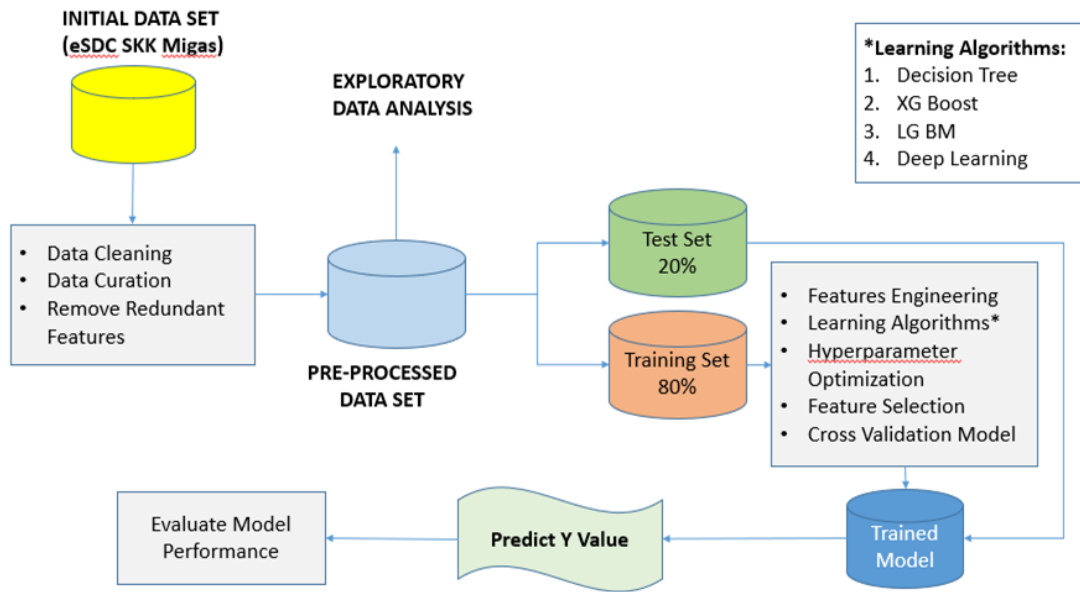


Figure 3.1. Project Process Workflow

### 3.1. Dataset & Features

In this project, we used the eSDC datasets, consisting of 21 original features. eSDC is subsurface datasets proprietary by SKK Migas. We used 60 - 40 proportion for train & valid+test data and 50-50 proportion for valid & test.

Those original features are :

Features Name	Explanation	Unit
temp	Field average temperature	F
depth	Field reservoir depth	feet
region	Region in Indonesia, separate as Sumatera, Kalimantan, Jawa, & Timur	-
field_name	Name of the fields	-
fluid	Fluid type, separate as Gas, Oil, and Gas-Oil	-
visc	Viscosity of the fluid	cp
NPV	Net Present Value	Million USD
location	Location of filed on the region, such as Jawa Timur	-

avg_fluid_rate	Field production daily rate	BOPD.e
total_cost	Total development cost	Million USD
opr_cost	Total operational cost	Million USD
cap_cost	Total capital cost	Million USD
saturate	Oil / Gas saturation in reservoir	fraction
api_dens	Fluid desity. For oil Oil-Gas it is API and Sg for Gas	-
perm	Permeability	md
poro	Porosity	fraction
inplace	Hydrocarbon initial in place	MMBO.e
project_level	Explaining project status on production stage	-
project_status	Project location: Onshore, Offshore, BOTH	-
PI	Profitability index	NPV/ total cost

### 3.2. Data Cleaning

	Total	Percent
temp	392	49.620253
depth	128	16.202532
region	56	7.088608
field_name	1	0.126582

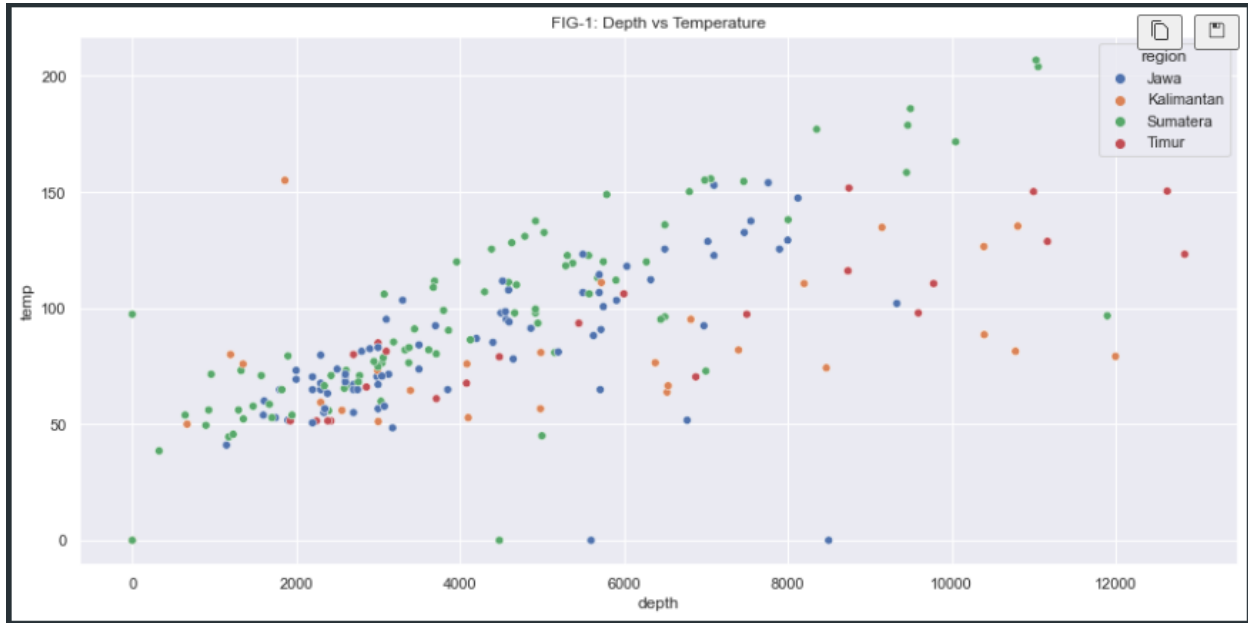
We have 4 major missing values. In oil and gas industry those top 3 variables are crucial because they are often related to development cost and then PI. To obtain a reliable model in PI prediction we decided to solve these issues

#### # Region

We notice that 56 region's data are missing, but when we look at those data we find it is difficult to tie the region by using another features. Despite missing values we decide to exclude those 56 data from dataset because they just represent around 7% of total. This deletion reveals some dilemmas that we are aiming to use as many as possible datas but we are bounded to classification problem that the model we are aiming to make is capable in making prediction in

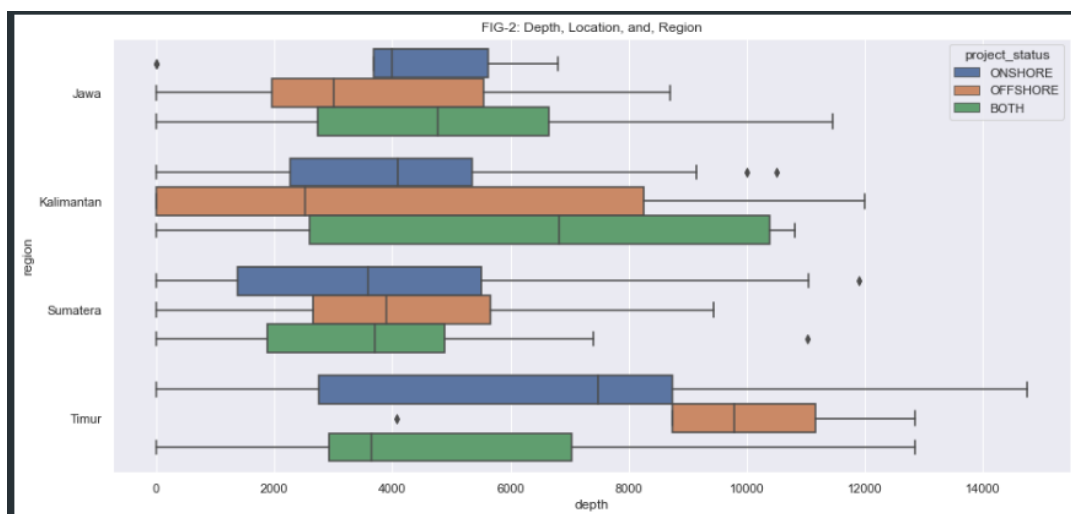
PI value for a certain region so we conclude that inputation in region could lead to model degradation and quality issue.

### # Depth



`Depth` has strong relation to `temp` as the drill string going deeper and so does the temperature. We can use this natural phenomena to predict temperature by using `depth`. As predictor, we have to solve `depth` at first to eliminate those missing datas, inconsistencies & outliers.

FIG-2 shows the depth distribution for indonesian oil & gas field (reservoir) categorized by their location and project status whether it is located on offshore or onshore or both. From FIG-2 we are informed that the depth could vary depending on its location and region.



On FIG-2 we notice that some anomalies exist and need to be solved. First issue relate to depth that has value == 0 which is impossible to have such shallow oil/ gas deposit so we will replace those values as well as NAN (second issue) by using region-project\_status clustered median. We believe median is suitable as replacement because based on business knowledge those value are consistent. Moving to eastern Indonesia we are expecting deeper sea and reservoir but not in Jawa-Kalimantan-Sumatera whose median values are quite similar.

### # Temperature

As mentioned above that `temp` has strong relation with `depth` lineary and this phenomenon has been known and confirmed by geological knowledge and petroleum physics so we are going to predict `temp` and doing inputation for missing value and 0 by using linear regression

Based on visual and theory representation, we believe the trend of `temp` will follow logarithmic shape so we decide to transform `depth` and `temp` to become logarithmic value

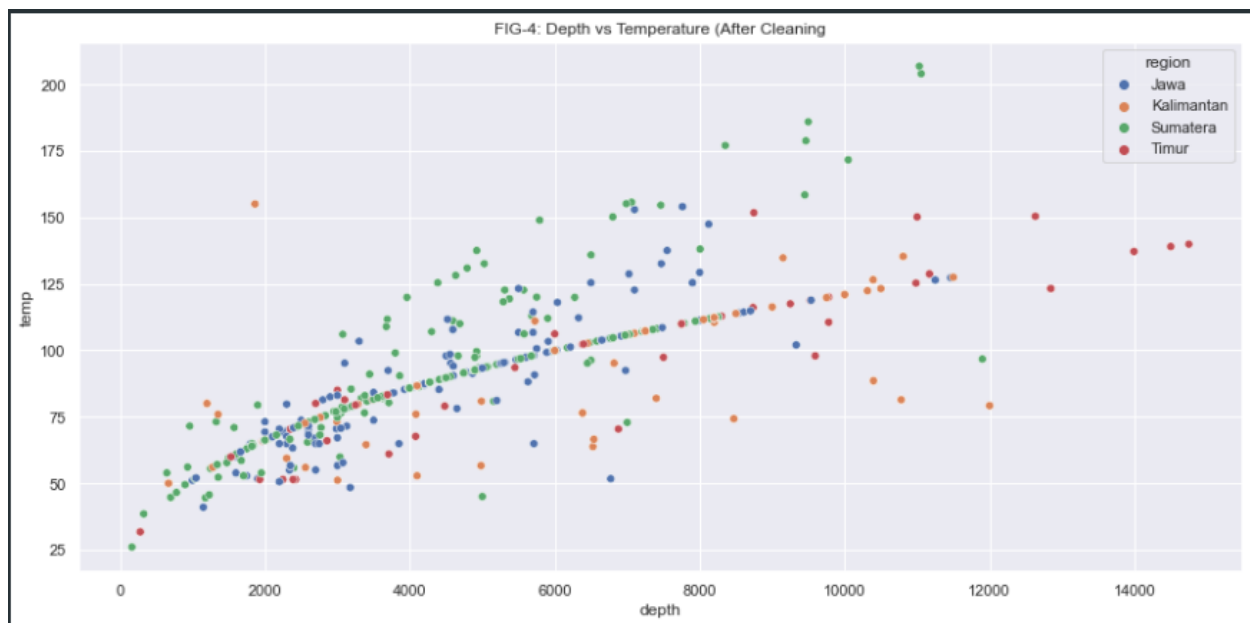
OLS Regression Results						
=====						
Dep. Variable:	temp	R-squared:	0.533			
Model:	OLS	Adj. R-squared:	0.532			
Method:	Least Squares	F-statistic:	295.1			
Date:	Sat, 02 Jul 2022	Prob (F-statistic):	1.02e-44			
Time:	14:56:17	Log-Likelihood:	8.3725			
No. Observations:	262	AIC:	-12.75			
Df Residuals:	260	BIC:	-5.608			
Df Model:	1					
Covariance Type:	HC3					
=====						
	coef	std err	z	P> z	[0.025	0.975]
-----						
Intercept	1.3489	0.181	7.469	0.000	0.995	1.703
depth	0.3742	0.022	17.178	0.000	0.331	0.417
=====						
Omnibus:	1.987	Durbin-Watson:	1.616			
Prob(Omnibus):	0.370	Jarque-Bera (JB):	1.796			
Skew:	-0.073	Prob(JB):	0.407			
Kurtosis:	3.379	Cond. No.	107.			
=====						
Notes:						
[1] Standard Errors are heteroscedasticity robust (HC3)						

As shown on summary tabel above we quite certain that the linear relationship can be modeled properly judged by some values that underlained linear regression foundation. Those values informed us that the model are the best fitted model available (refer : Gauss-Markov)

Those values are:

- Durbin watson = Near 2, which is suitable to conclude that our model are not performing some serious autocorrelation
- F-stat (prob) =  $1.02e-44$ , meaning `depth` has strong effect on temperature together with the intercept ( $< 0.05$ )
- R-square = 0.532, meaning we can model/ explain more that 50% variance in `temp` by using `depth` alone
- Prob (JB) = 0.407, meaning our model's residual are normaly distributed ( $> 0.05$ ) and not performing some heteroskedastic
- Multicollinearity = Not tested, but we can conclude this effect is not coming into play because we just use single `depth` variable

After cleaning process is done we obtained better temperature dataset and its relation to depth as shown below:



### # Field Name

We notice that 1 field doesn't have field name but we decide to input this missing value in field\_name by using dummy name as PETROGAS\_FIELD.



### 3.3. Feature Engineering

#### # PI Engineering

```
project_level
E0. On Production          2.033117
E1. Production on Hold     3.128682
E2. Under Development      2.420781
E3. Justified for Development 0.849384
E4. Production Pending     0.965692
E5. Development Unclarified 1.459044
E6. Further Development    1.223655
E7. Production Not Viable  0.749261
E8. Further Development Not Viable 2.094478
X0. Development Pending    1.240810
X1. Discovery under Evaluation 1.220731
X2. Development Undetermined 0.850525
X3. Development Not Viable  0.584434
Name: PI, dtype: float64
```

Based on dataset that we have, we agree that those PI values ranging in numerical manner. We believe that having this number on high level evaluation/ predictive model output are not informative so we decide to categorize those elements by using unique name related to their prospect as : Pros and NonPros. To simplify the modeling process we replace those value to become : Pros = 1 and NonPros = 0.

The naming process begins by doing some analysis to find mean value for each `project\_level`. We believe that `project\_level` has strong relation to `PI` because we can spot the differences on their mean related to their level. For instance, E0 when fields are categorized as "On Production" meaning those field has already produced and delivered a cashflow (projects are profitable) has mean `PI` value as high as 2 (profitable). We agree that this visual segmentation process is prone to bias because it is not supported by statistical analysis related to their dataset.

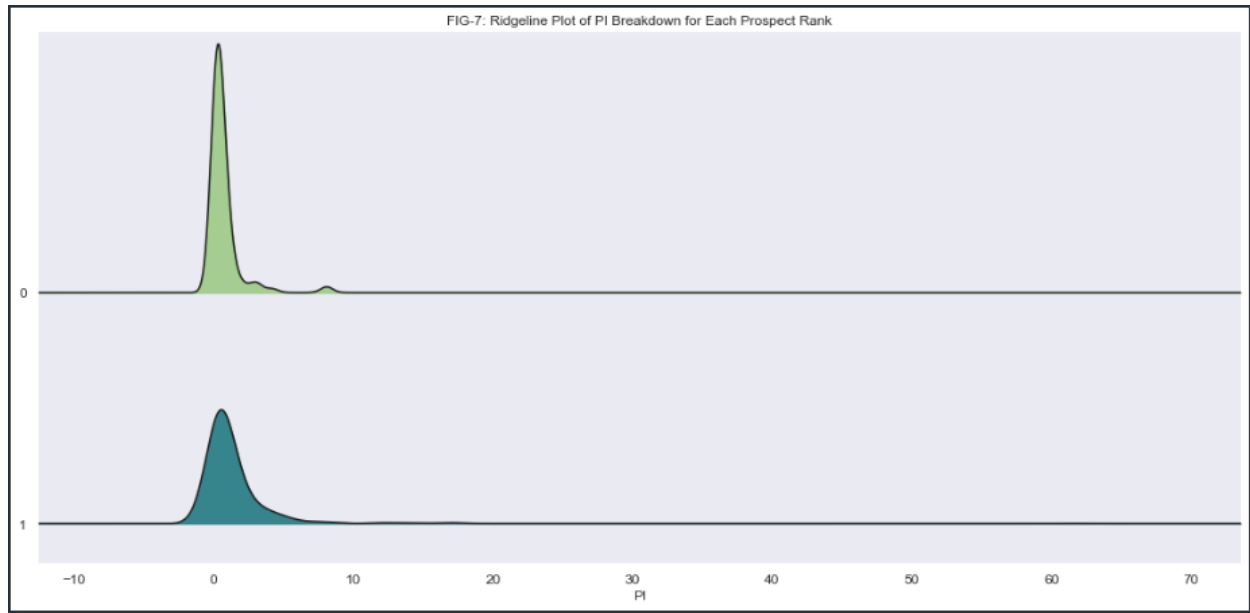
To make an objective studies and deliver a reliable model we are going to use statistical analysis to support out segmentation process in separating `PI` adjacent to their `project\_level` by assuming their distribution are normal distribution (FIG-5). This process begin by doing statistical significant analysis sequentially as follow:

- Separate Pros + NonPros (is it Pros significant by mean value from NonPros?)

Statistical significant test will be conducting using t-test. In order to perform such test we will generate new feature to accomodate prospect rank as mentioned above (Pros=1 and NonPros=0), this feature will be called `prospect\_rank`

That feature are related to `project\_level` segmented by their `PI` mean value. For clarity we show the segementation in detail just below:

- Pros = 1 --> E0, E1, E2, E8, E3, E4, E5, E6, X0, X1, X2
- NonPros = 0 --> E7 & X3



From joyplot on FIG-7 above we can spot the differences between those 2 data distribution when single mean value might be not representative to be used on statistical significant analysis (independence) as each of them having fat tailed distribution. To overcome this issue we are going to conduct multiple significant test known as t-Test, ChiSquared Test, f-Test, and The Kruskal-Wallis H-Test.

- T-test:

This is a test for the null hypothesis that 2 independent samples have identical average (expected) values. This test assumes that the populations have identical variances by default

- Separating 1 from 0

```
Ttest_indResult(statistic=4.351033480151706, pvalue=1.551286295864342e-05)
```

If we assumed that p-value cut-off to infer conclusion (rejecting null) from data as high as 0.05 (5%) then from our calculation shown on code above we can conclude that from the data we

have and segmentation we did between 1 & 0, we are able to separate 1 as it has independencies (stastically speaking) from 0.

- Chi-Squared Test:

This is a test for the null hypothesis that 2 independent samples. One of the least known applications of the chi-squared test is "testing the similarity between two distributions". If the two distributions were the same, we would expect the same frequency of observations in each bin. Please remember, this is a test for whole distribution shape not just mean value as t-Test did to conclude whether 2 datas along with their distribution are different statistically.

The reason lies in the fact that the two distributions have a similar center but different tails and the chi-squared test tests the similarity along the whole distribution and not only in the center, as we were doing with the previous tests

- Separating 1 from 0

```
Power_divergenceResult(statistic=6745.396187513456, pvalue=0.0)
```

If we assumed that p-value cut-off to infer conclusion (rejecting null) from data as high as 0.05 (5%) then from our calculation shown on code above we can conclude that from the data we have and segmentation we did between 1 & 0, we are able to separate 1 as it has independencies (stastically speaking) from other.

- F-test:

With multiple groups, the most popular test is the F-test. The F-test compares the variance of a variable across different groups. This analysis is also called analysis of variance, or ANOVA.

The ANOVA test has important assumptions that must be satisfied in order for the associated p-value to be valid:

- The samples are independent.
- Each sample is from a normally distributed population.
- The population standard deviations of the groups are all equal. This property is known as homoscedasticity.

```
F_onewayResult(statistic=8.007395318819713, pvalue=0.004786252973762854)
```

The f-Test p-value is less than 0.05 (5%), implying a strong rejection of the null hypothesis of no differences in the `PI` distribution across `prospect\_rank`.

- The Kruskal-Wallis H-test:

Because the last assumption of f-Test are not fulfilled we add last test (The Kruskal-Wallis) to confirm independencies

The Kruskal-Wallis H-test tests the null hypothesis that the population median of all of the groups are equal. It is a non-parametric version of ANOVA. The test works on 2 or more independent samples, which may have different sizes.

```
KruskalResult(statistic=23.3758812435488, pvalue=1.3323867878050264e-06)
```

Still, The p-value is significantly less than 0.05 (5%), implying a strong rejection of the null hypothesis of no differences in the `PI` distribution across `prospect\_rank`.

### **# Operatorship Engineering**

We don't need all operators name on this analysis because we believe it will not deliver significant information regarding field economic. Instead, we are going to replace those name by Pertamina & Non-Pertamina to separate NoC and others.

### **# Features Dropping**

Based on analysis that has been done we will drop some features that are not relevant anymore for further evaluation and modeling. Those features were critical when we modeled the `depth`, `temp`, `prospect\_rank` and performed statistical independence test. Those features are :

- `field\_name`
- `project\_level`
- `cap\_cost`
- `opr\_cost`
- `total\_cost`
- `NPV`
- `PI`

### **# Unit Conversion**

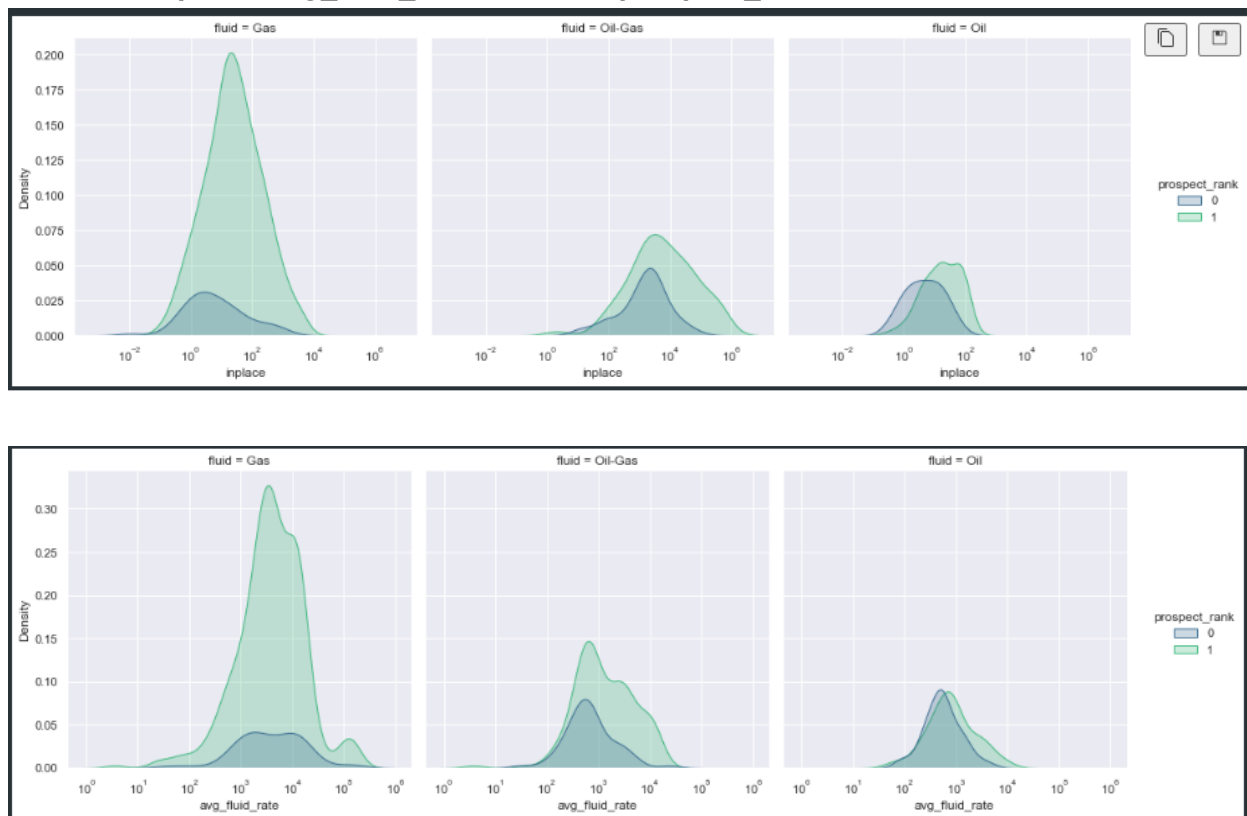
We will modified some features unit to make it more representative and consistent. Below is the list of those features and their unit:

- inplace (Oil in MSTB, Gas in BSCF, and Oil\_Gas in MSTB), will be modified to MMBO.E
- depth (feet)
- temp (fahrenheit)
- poro (dimensionless)

- perm (md)
- saturate (dimensionless)
- api\_dens (Oil in API, Gas in Specific Gravity, Oil\_Gas in API), no modification is needed since the both unit are dimensionless
- visc (cp), gas visc are negligible (zero) while oil is in centipoise (cp)
- avg\_fluid\_rate (Oil in BOPD, Gas in MMSCFD, and Oil\_Gas in BOPD), will be modified to BOPD.E [12862858643001](#)

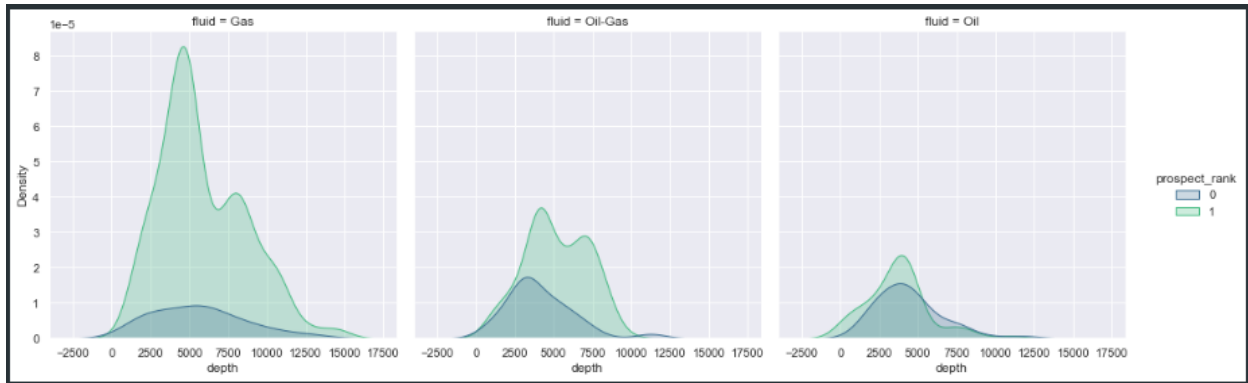
### 3.4. Exploratory Data Analysis

#### # Effect of inplace, avg\_fluid\_rate & fluid on prospect\_rank

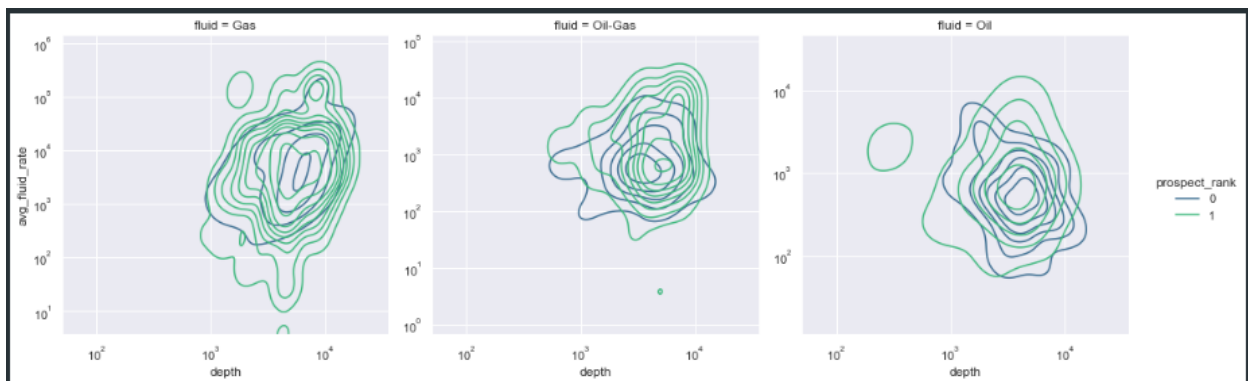
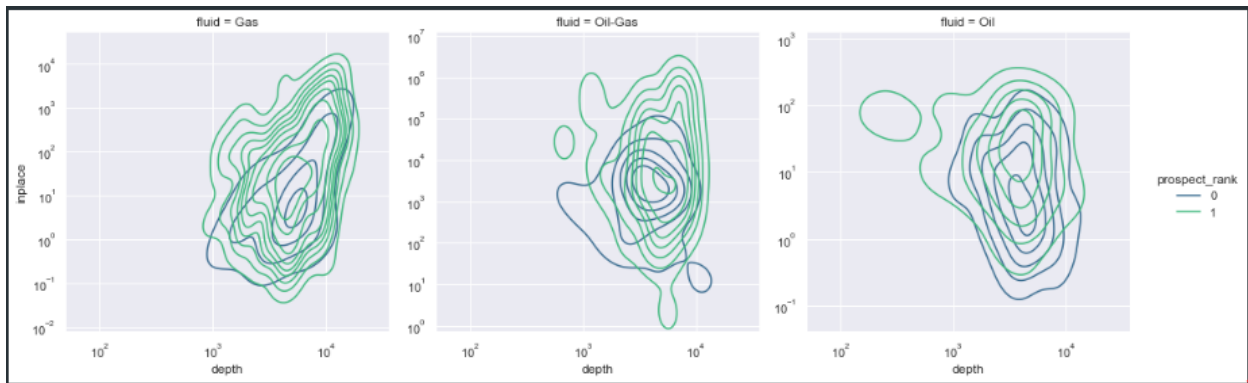


We suspect there is a causal effect of 'inplace' & 'avg\_fluid\_rate' in 'prospect\_rank'. While we segmented on 'fluid' we notice that by having bigger 'inplace' and 'avg\_fluid\_rate' could lead us to more preferable economic of scale benefit therefore significantly improve our field 'prospect\_rank' but we notice a high level anomaly on 'fluid' == 'Gas' where 'avg\_fluid\_rate' are not giving significant impact any more on 'prospect\_rank'. This issue could be inferred from graph when 'prospect\_rank' == 0 exists together along 'prospect\_rank' 1. This issue need to be addressed in more detail analysis during modeling whether additional features are needed to clarify or not.

### # Effect of depth, inplace, rate, and fluid on prospect\_rank



On above plot, we see minor insight regarding `depth` and its effect on `prospect\_rank` while data is segmented by `fluid`. We can infer from high level that neither `fluid` nor `depth` will give significant influences on field prospect, instead most of the field will have better chance to be more economical when `depth` is shallower. Probably, we have to classify the data further by looking to other basic features to get better `prospect\_rank` classification.



More insightful analysis is obtain when we classify `depth` further by using `inplace` and `avg\_fluid\_rate`. From 2 last plot above we spot meaningful informations that `inplace` and `avg\_fluid\_rate` play significant role in determining field `prospect\_rank`. When fields are having

a deeper reservoir target they need either bigger `inplace` or `avg\_fluid\_rate` in order to become more economical.

We agree that some anomalies exists, for example on `Gas` where `prospect\_rank` are not clasify properly. We believe, although more analysis in ML are needed, that this issue occurs only on `Gas` because sometimes gas production is limited by market potential and could limit their withdrawal rate. This problem could erode economical value in accordance. So, we suspect on `Gas` that either `region` or `location` could play major role to spot non market potential area/ region.

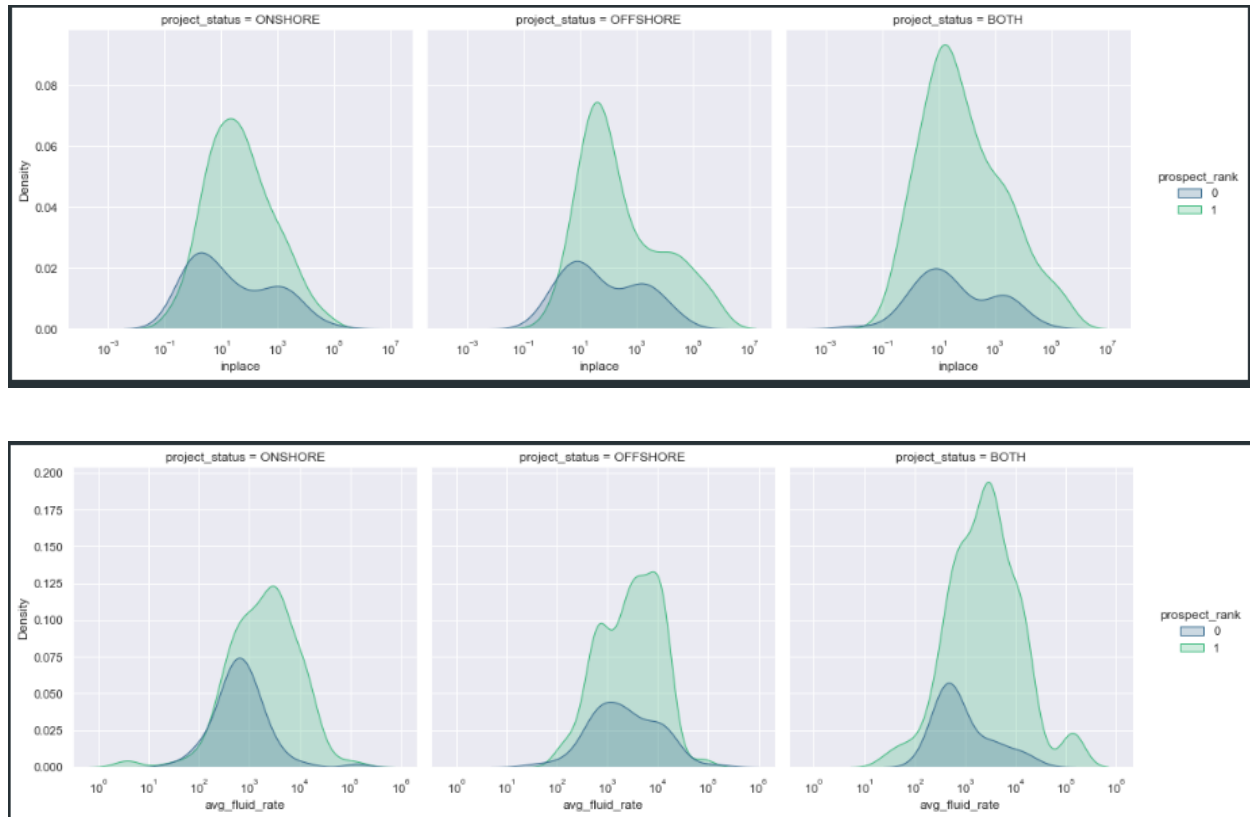
### # Effect of inplace, rate, region, and fluid==Gas on prospect\_rank



As we spot earlier on previous analysis that `region` plays a crucial role masking `prospect\_rank` clasification using `depth`, `inplace`, and `avg\_fluid\_rate` when `fluid`==Gas. On above plots, we are convinced that 1 region (timur) is probably the reason causing this issue. When all others region shows consistent trend reflecting good and rational relationship between `depth`, `inplace`, `avg\_fluid\_rate` on `prospect\_rank`, but `region`==Timur shows the opposite. This information/ our hypothesis from data relates to actual condition where market

potential for gas on eastern indonesia (timur) is very limited, therefore a good relationship between `depth`, `inplace`, `avg\_fluid\_rate` on `prospect\_rank` are not shown clearly.

### # Effect of inplace, rate, and Project\_status (location) on prospect\_rank

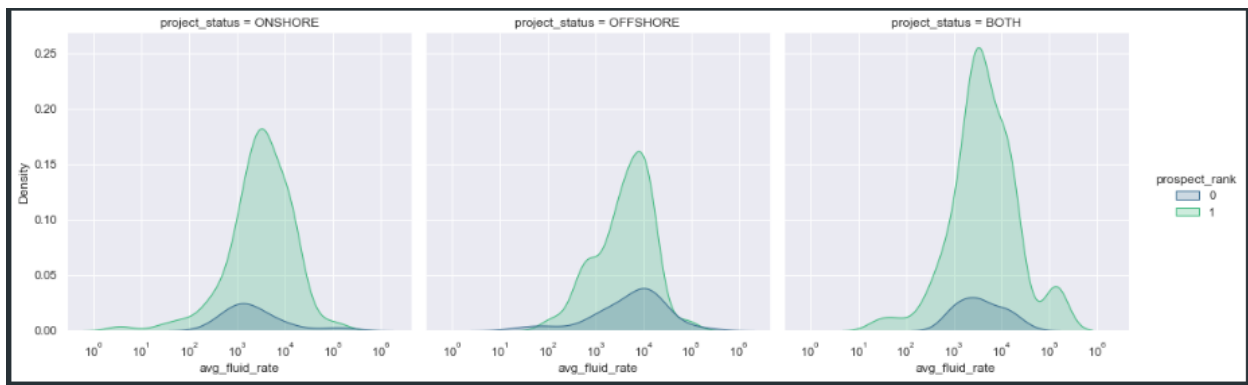
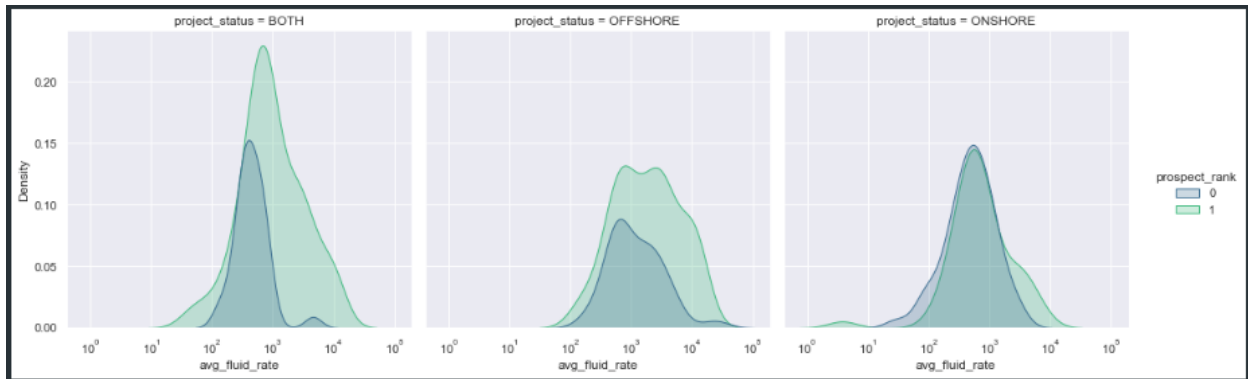


On two plots above we spot many insights that explain how both `inplace` and `avg\_fluid\_rate` could be a valuable predictor:

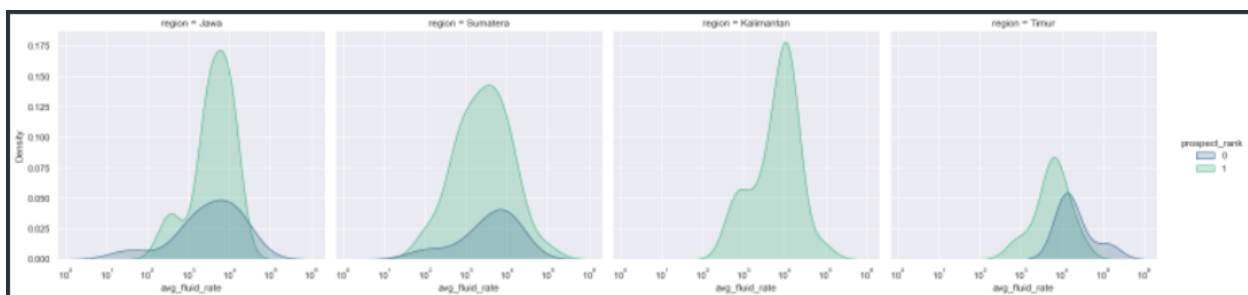
- First, we notice that when we have bigger `inplace` it improve our chance to have better `prospect\_rank` whenever our field is located on offshore/ onshore/ both for all fluid type.
- Second, `avg\_fluid\_rate` plays significant role since it can reveal very insightful notions that by having higher rate we are inclined to have better economic value for all project location and fluid type.
- Third, `avg\_fluid\_rate`  $\geq 1000$  BOPD.E could be considered as fundamental rate cut off, whenever we pass this value we are inclined to have profitable field but some concern remains that on `project\_status`==Offshore, probably, we need additional variable than single `avg\_fluid\_rate` in order to spot a better cut off as minimum hydrocarbon fluid rate.

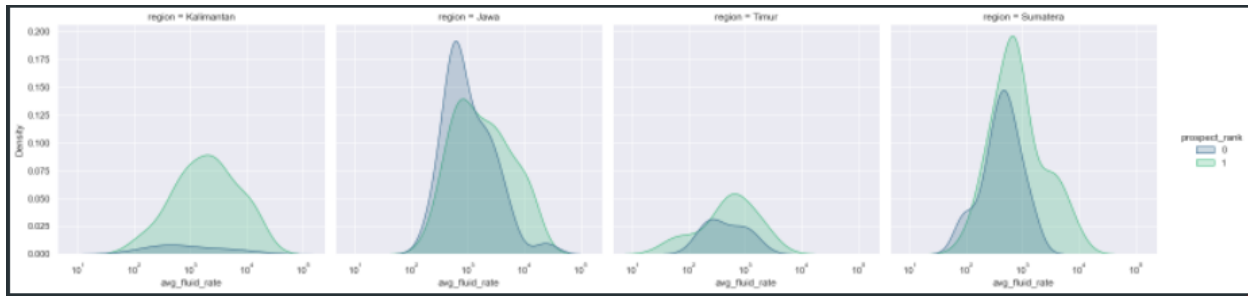
So, when we have a new field, by ensuring this field to deliver significant high rate (above 1000 BOPD.E) might lead us to get a better chance having profitable oil/ gas field regardless their location and fluid type.





Looking forward on more detailed analysis after figuring out that `avg_fluid_rate` could be used as strong predictor explaining `prospect_rank` when rate is above 1000 BOPD.E, although showing minor effect on offshore location for gas only fluid type (shown on 2 plots above), we intended to expand the analysis by breaking down offshore projects according to their `fluid`==Gas and their `region`, as shown on plot below:

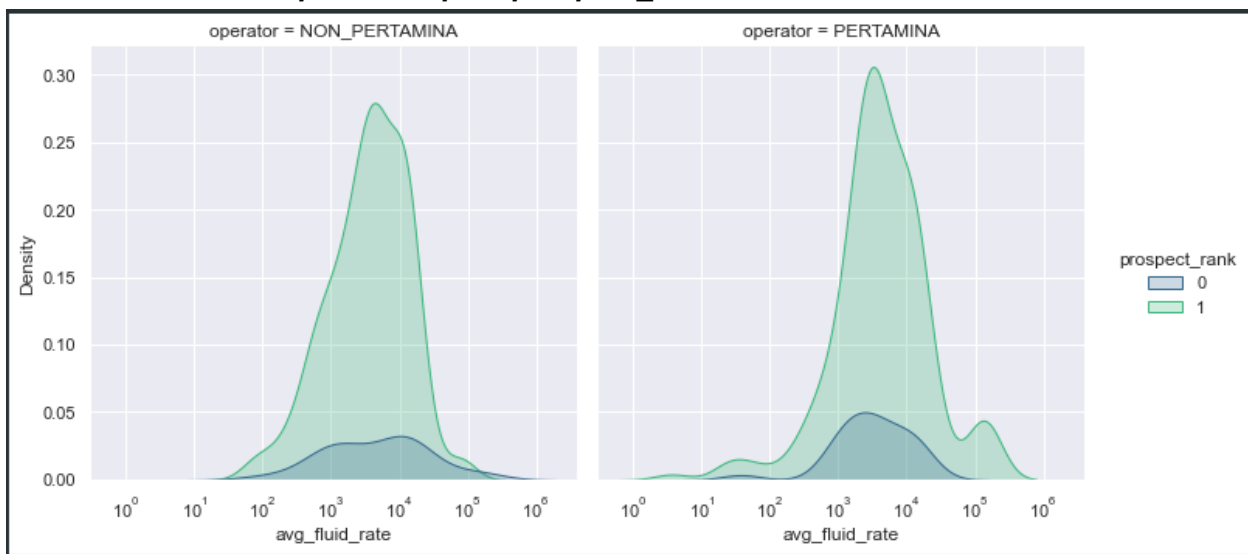


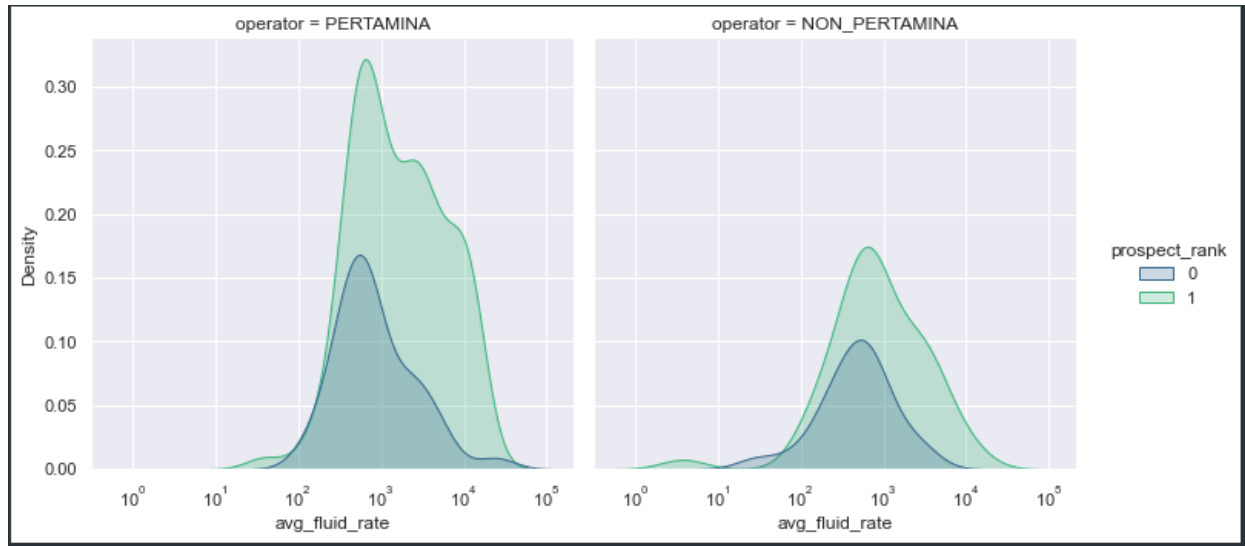


We obtained many meaningful informations by using 2 plots above:

- Kalimantan is the most preferable `region` to produce hidrocarbon on offshore, whenever the `avg\_fluid\_rate`, We might have profitable oil & gas field, especially when `avg\_fluid\_rate` > 1000 BOPD.E.
- `avg\_fluid\_rate` >= 1000 BOPD.E could be used as profitable economic cutoff for `fluid`==Oil & `fluid`==Oil&Gas and any kind of regions and locations (second plot above)
- Beside Kalimantan, all gas field projects on offshore have strong tendency to be less profitable when those fields could produce over 1000 BOPD.E. It might be an early indication that those regions have market/ infrastructure issues related to gas. These issues are important and deserve more troughout analysis on modeling.

### # Effect of rate, and operatorship on prospect\_rank





Those 2 plots above imply that:

- Since Pertamina has more diverse field than others it might improve Pertamina's portfolio
- When it comes to gas field, more projects have better profitability under Pertamina than others. This condition might exist probably because of project integration, market readiness, and sound infrastructure alignment inside Pertamina business unit
- On oil project, neither Pertamina nor Others could deliver better performance on profitability but we might use ``avg_fluid_rate` > 1000 BOPD.E` as economic cutoff.

### **General Conclusion from EDA:**

1. We suspect there is a causal effect of ``inplace`` & ``avg_fluid_rate`` in ``prospect_rank``. While we segmented on ``fluid`` we notice that by having bigger ``inplace`` and ``avg_fluid_rate`` could lead us to more preferable economic of scale benefit
2. When fields are having a deeper reservoir target they need either bigger ``inplace`` or ``avg_fluid_rate`` in order to become more economical.
3. We suspect on ``Gas`` that either ``region`` or ``location`` could play major role to spot non market potential area/ region.
4. When all others region shows consistent trend reflecting good and rational relationship between ``depth``, ``inplace``, ``avg_fluid_rate`` on ``prospect_rank``, but ``region`==Timur` shows the opposite. This information/ our hypothesis from data relates to actual condition where market potential for gas on eastern indonesia (timur) is very limited, therefore a good relationship between ``depth``, ``inplace``, ``avg_fluid_rate`` on ``prospect_rank`` are not shown clearly.
5. When we have a new field, by ensuring this field to deliver significant high rate (above 1000 BOPD.E) might lead us to get a better chance having profitable project regardless their location if fluid type is oil but it might not work for gas if it is located at offshore.
6. Kalimantan is the most preferable ``region`` to produce hidrocarbon on offshore, whenever the ``avg_fluid_rate``, We might have profitable project, especially when ``avg_fluid_rate` > 1000 BOPD.E`.

7. ``avg_fluid_rate` >= 1000 BOPD.E` could be used as profitable economic cutoff for ``fluid`==Oil` & ``fluid`==Oil&Gas` on any kind of regions and location.
8. Beside Kalimantan, all gas field projects on offshore have strong tendency to be less profitable when those fields could produce over 1000 BOPD.E. It might be an early indication that those regions have market/ infrastructure issues related to gas.
9. Since Pertamina has more diverse field than others it might improve Pertamina's portfolio
10. When it comes to gas project, more projects have better profitability under Pertamina than others. This condition might exist probably because of project integration, market readiness, and sound infrastructure alignment inside Pertamina business unit
11. On oil project, neither Pertamina nor others could deliver better performance on profitability but we might use ``avg_fluid_rate` > 1000 BOPD.E` as economic cutoff.

### 3.5. Modeling Process

#### **# Data Transformation**

We are not going to apply any transformation on numerical data but for categorical we apply `one_hot_encoding`.

#### **# Modeling**

In the modeling process we train several models i.e :

- Logistic Regresion
- Random Forest
- Decision Tree
- KNN Classifier
- LGBM
- XGBoost
- SVC
- Gaussian Naive Bayes

#### Gini Performance Evaluation

```
Logistic Regression Gini : 0.38987024665981496
Random Forest Gini      : 0.9997430626927031
Decision Tree Gini      : 0.9997430626927029
KNN Classifier Gini     : 0.6619186793422405
LGBM Gini               : 0.9997430626927026
XGBoost Gini            : 0.9997430626927031
SVC Gini                : 0.18997302158273377
GNB Gini                : 0.41408658787255925
```

We chose the three (3) initial models that had the highest Gini score and as can be seen from the figures above where it was found that the best models were Decision Tree, LGBM, and XGBoost.

Based on baseline model analysis, after conducting random search (hyperparameter tuning) we select 1 model from 3 selected models that are top performers for more advanced tuning using HyperOpt. That model is XGBoost.

**NOTE :** We find there is no significant improvement by using DL (demonstrated on these branch & another branch that apply DL using grid search) and since we are convinced that for multi features/ tabular data when all those features are heterogeneous (not homogenous like pictures/ videos dataset) XGBoost might be the best possible option. Moreover, for simplification since time is our primary concern beside accuracy, keeping XGBoost as main model and doing some advanced optimization method (HyperOPT) to search an optimal configuration is considered as the best decision.

<https://www.quora.com/Why-is-XGBoost-among-most-used-machine-learning-method-on-Kaggle>

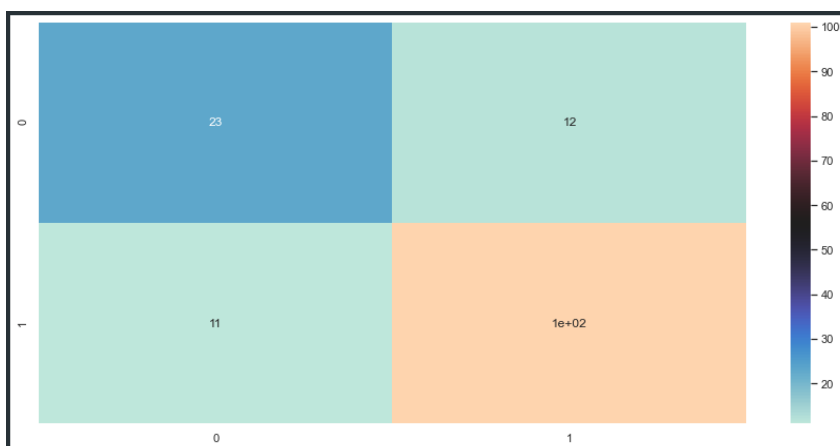
### # Model Result on Test Data

With xgboost as our model, we retrained the model with train and valid data and tested it on the test data. Our test results can be seen in the figure below.

- Test Data Result

```
Test Data: accuracy_score: 0.8435374149659864
Test Data: balanced_accuracy_score: 0.7794642857142857
Test Data: precision_score: 0.8938053097345132
Test Data: recall_score: 0.9017857142857143
Test Data: roc_auc_score: 0.7926020408163266
```

- Confusion Matrix



Stopping criteria in our Hyperopt tuning is based on its reability in term of acceptable roc\_auc\_score ~0.80, high recall\_score ~0.90 which is suitable to identify prospective fields as many as possible to be developed, high precision\_score ~0.89 obtained relative to recall\_score, therefore, could help users to find real/ true prospective fields in timely manner with minimal re-evaluation job.

## 4. Product

### 4.1. User Persona

This app is suitable for Engineers, Development Team, and New Venture Companies who seek to optimize their work in term of field profitability screening. By using this tool, users are enable to focus on field that has higher chance to become profitable. Furthermore, both development team and new ventures can utilize this tools to expand their portofolio.

### 4.2. Product Features

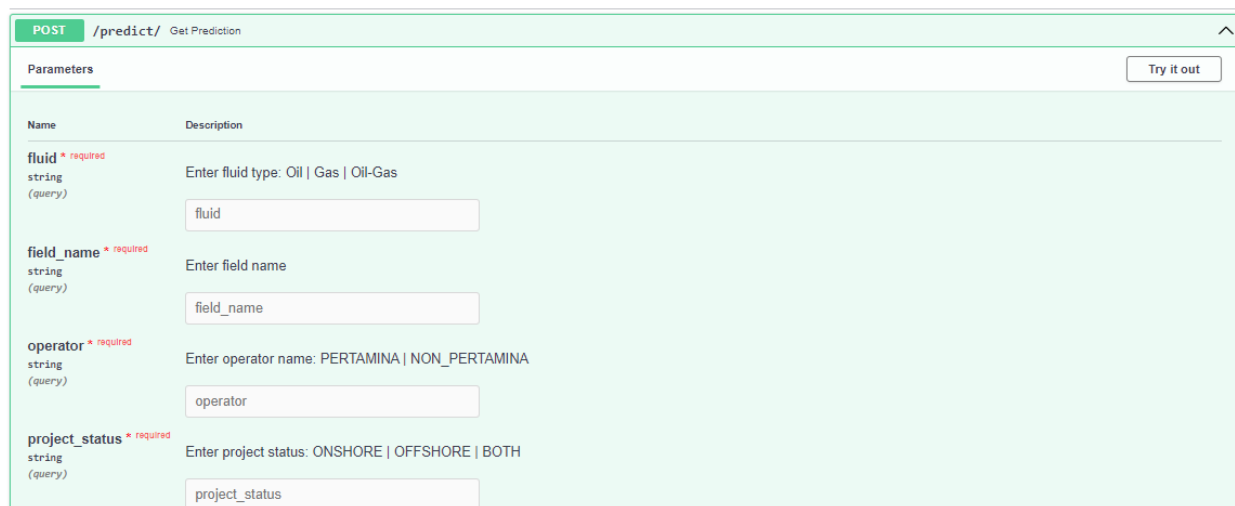
From the model that we have created, we save it with the Joblib module in the form of pkl. Next we make a deployment using heroku. The results of our deployment can be seen on the link <https://peaceful-ravine-66425.herokuapp.com/docs>. The snapshot of the interface of our product is as shown in the image below

#### Field Profitability Index Prediction 1.0 OAS3

/openapi.json

Predict the field profitability using its nearby field information

default



The screenshot displays the Swagger UI for the 'Field Profitability Index Prediction' API. The interface is titled 'default' and shows a 'POST' method for the endpoint '/predict/'. The description is 'Get Prediction'. Below the endpoint, there is a 'Parameters' section with a 'Try it out' button. The parameters are listed in a table with columns 'Name' and 'Description'.

Name	Description
<b>fluid</b> * required string (query)	Enter fluid type: Oil   Gas   Oil-Gas <input type="text" value="fluid"/>
<b>field_name</b> * required string (query)	Enter field name <input type="text" value="field_name"/>
<b>operator</b> * required string (query)	Enter operator name: PERTAMINA   NON_PERTAMINA <input type="text" value="operator"/>
<b>project_status</b> * required string (query)	Enter project status: ONSHORE   OFFSHORE   BOTH <input type="text" value="project_status"/>

The application consists of a form that contains several sections that need to be filled out. the contents of the form will be calculated by the model that has been made to get the profitability index and its probability.

The parts that need to be filled in include:

- fluid

Users need to enter fluid type in form of : Gas, Oil, Oil-Gas

- field\_name

This field could be left blank or filled with field name

- operator

Users need to define field operator from 2 options : PERTAMINA or NON\_PERTAMINA

- project\_status

Users need to define field operator from 3 options : ONSHORE, OFFSHORE, BOTH

- inplace

Users need to input estimated field inplace (structure) in MMBO.e

- depth

Users need to input estimated field depth in feet

- temp

Users need to input estimated field temperature in fahrenheit

- poro

Users need to input estimated field porosity in fraction

- perm

Users need to input estimated field permeability in md

- saturate

Users need to input estimated field hydrocarbon saturation in fraction

- api\_dens

Users need to input estimated field fluid density for oil & oil-gas and gas. When it comes to gas the unit is in SG otherwise is in API

- visc

Users need to input estimated field viscosity in cp

- avg\_fluid\_rate

Users need to input estimated field production rate (structure) in BOPD.e

- location

Users need to input field location in: Aceh | Jambi | Jawa Barat | Jawa Tengah | Jawa Timur | Kalimantan Selatan | Kalimantan Tengah | Kalimantan Timur | Kalimantan Utara | Laut Cina Utara | Laut Jawa | Laut Natuna | Laut Natuna Utara | Laut Seram | Laut Timor | Maluku | Papua Barat | Riau | Selat Makasar | Selat Malaka | Sulawesi Barat | Sulawesi Selatan | Sulawesi Tengah | Sulawesi Tengah (offshore) | Sumatera Barat | Sumatera Selatan | Sumatera Utara | Teluk Berau

- region

Users need to input field region in: Jawa | Kalimantan | Sumatera | Timur

### 4.3. Result

When we have finished filling in all the data on the form we can click submit and the application will calculate the results whether our field is profitable (1) or marginal (0) as long as its probability.

#### # Profitable Field Result

Response body

```
{"result": "1| Profitable", "proba": [0.997511625289917]}
```

#### # Marginal Field Result

Response body

```
{"result": "0| Marginal", "proba": [0.9823238849639893]}
```



## 5. Conclusion

1. XGBoost has been selected as model ready and has been deployed on Heroku
2. Decision to select XGBoost is based on its reability in term of acceptable roc\_auc\_score ~0.80, high recall\_score ~0.90 which is suitable to identify prospective fields as many as possible to be developed, high precision\_score ~0.89 obtained relative to recall\_score, therefore, could help users to find real/ true prospective fields in timely manner with minimal re-evaluation job, furthermore, on its capability to allow engineer doing iteration, finetuning and optimization as fast as possible
3. Information from trees in XGBoost combined with features\_importances could help engineer getting meaningful insight related to intrinsic factors that could effect field prospective index. Also, by using these trees, engineer could weight their focus on certain aspects to increase their odd finding the profitable oil & gas assets

## 4. References

- eSDC Dataset

## 5. Github Links

[https://github.com/aprds/field\\_profitability\\_index](https://github.com/aprds/field_profitability_index)

## 6. App Links

<https://peaceful-ravine-66425.herokuapp.com/docs>