PROJECT REPORT

Topic: Predicting Life Expectancy using Machine Learning

Submitted by:

Toyada Pattanaik
Silicon Institute of Technology

CONTENT

1 INTRODUCTION

1.1 Overview

This is a typical Regression Machine Learning project that leverages historical data to predict insights into the future. This project is aimed at predicting Life Expectancy rate of a country given various features.

Life expectancy is a statistical measure of the average time a human being is expected to live, Life expectancy depends on various factors: Regional variations, Economic Circumstances, Sex Differences, Mental Illnesses, Physical Illnesses, Education, Year of their birth and other demographic factors. The project provides a way to predict average life expectancy of people living in a country when various factors such as year, GDP, education, alcohol intake of people in the country, expenditure on healthcare system and some specific disease related deaths that happened in the country are given.

1.2 Purpose

The project aims to train a machine learning model on a set of historical data in order to predict the life expectancy of a country when the various parameters affecting it are changed or randomized. It enables the user to input values for different variables and visualize how it affects the life expectancy.

2 LITERATURE SURVEY

2.1 Existing Problem

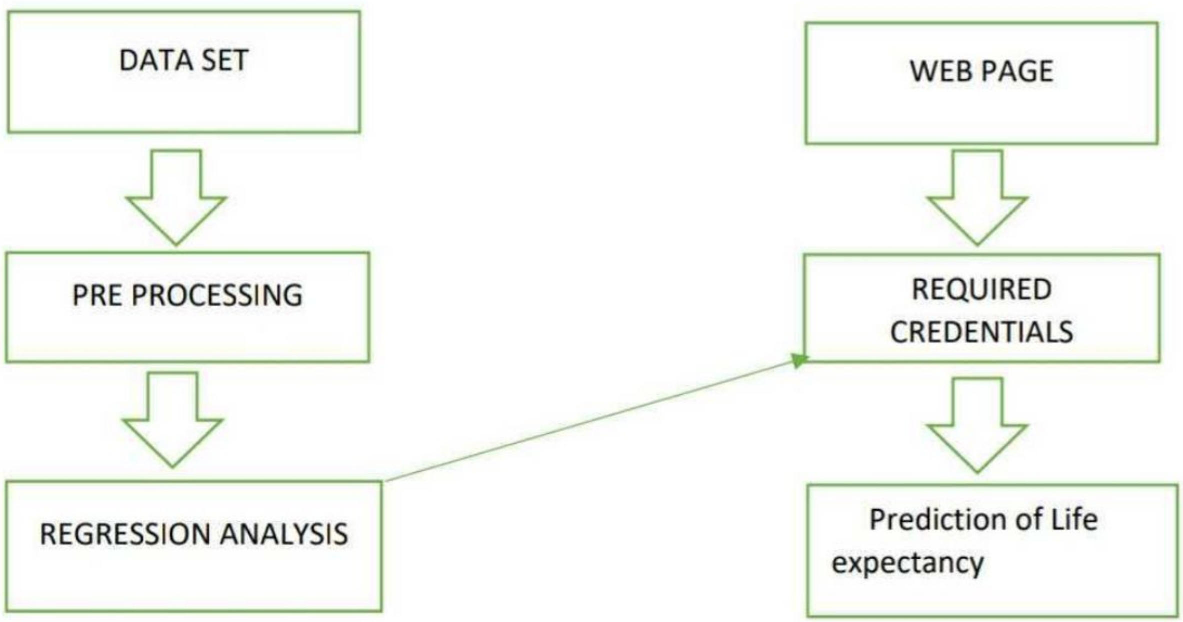
Life expectancy is a statistical measure of the average time a human being is expected to live, Life expectancy depends on various factors: Regional variations, Economic Circumstances, Sex Differences, Mental Illnesses, Physical Illnesses, Education, Year of their birth and other demographic factors. We need to know how one specific factor among these affects life expectancy.

2.2 Proposed Solution

Our project aims to solve this problem by performing Regression Analysis on the historical dataset provided by WHO. It will enable us to understand how life expectancy depends on the various factors specified in the dataset. It will be beneficial so as to direct our focus on the factors which greatly affect life expectancy.

3 THEORETICAL ANALYSIS

3.1 Block Diagram



3.2 Hardware/Software Designing

To complete my project, I have used,

- IBM Cloud
- IBM Watson
- Node-Red

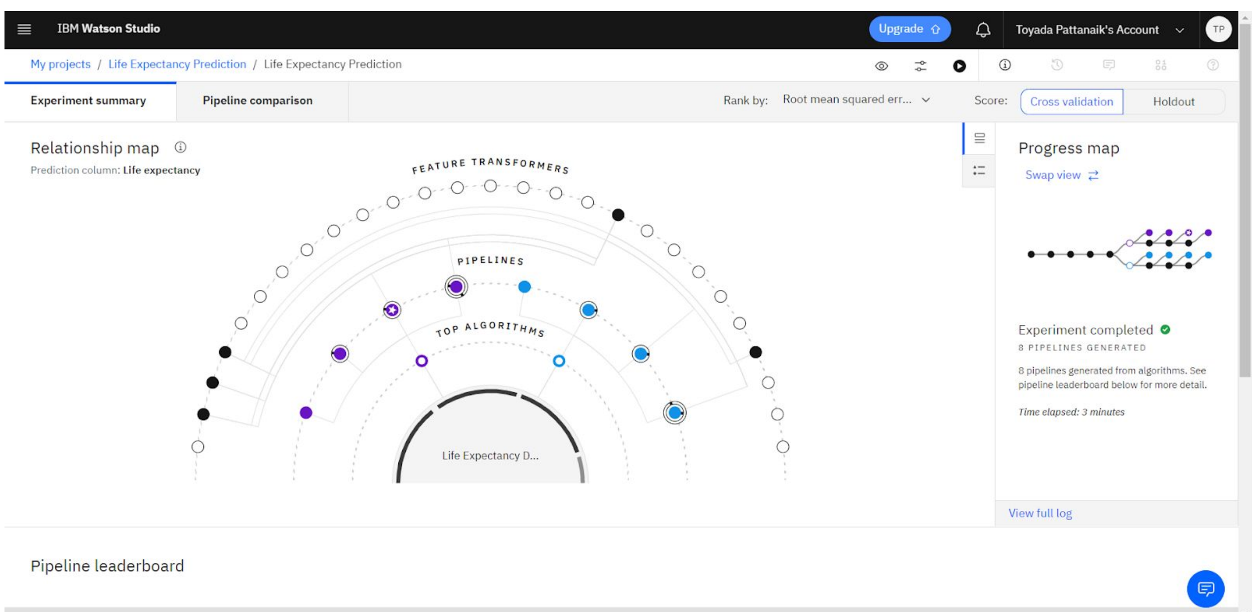
IBM Cloud was used to store the dataset and the machine learning services used in the project. Regression Analysis was carried out using the Auto AI feature of IBM

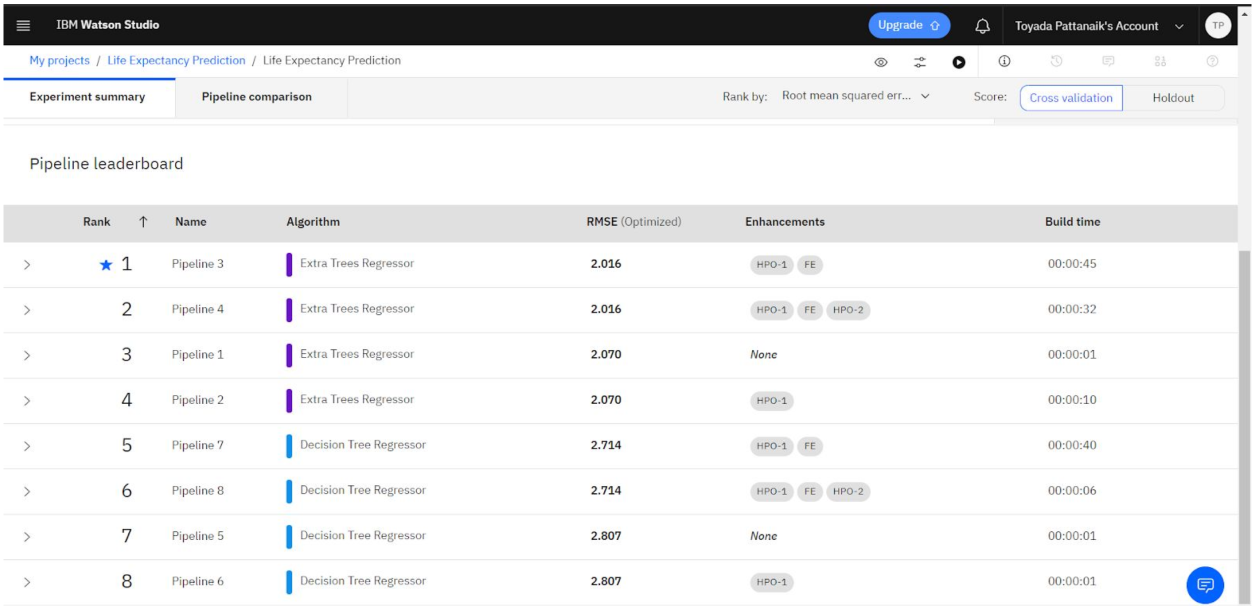
Watson Studio. Node-Red, based on JSON, was used to create the web page which integrated the machine learning model to show results for the user inputted values.

4 EXPERIMENTAL INVESTIGATIONS

We carried our Regression Analysis on the dataset in order to determine the dependency of Life Expectancy on various factors. In statistical modeling, regression analysis is a set of statistical processes for estimating the relationships between a dependent variable and one or more independent variables.

Regression analysis was performed using the Auto AI feature in the Watson Studio which helps automate the coding process.





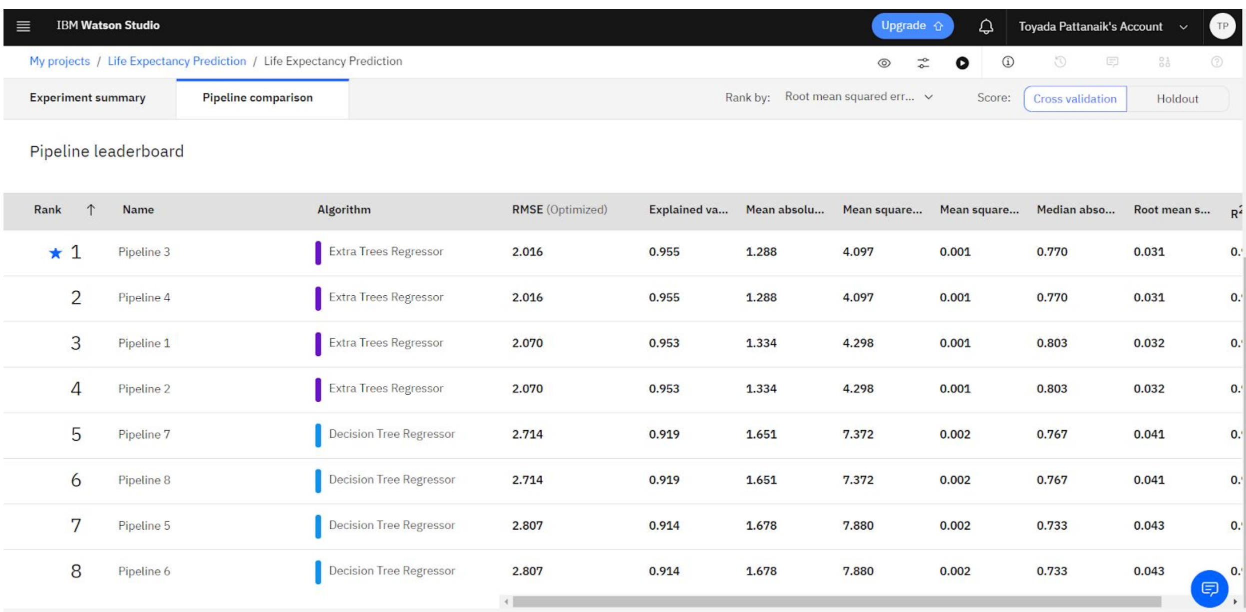
IBM Watson Studio | Upgrade | Toyada Pattanaik's Account | TP

My projects / Life Expectancy Prediction / Life Expectancy Prediction

Experiment summary | Pipeline comparison | Rank by: Root mean squared err... | Score: Cross validation | Holdout

Pipeline leaderboard

Rank	Name	Algorithm	RMSE (Optimized)	Enhancements	Build time
★ 1	Pipeline 3	Extra Trees Regressor	2.016	HPO-1 FE	00:00:45
2	Pipeline 4	Extra Trees Regressor	2.016	HPO-1 FE HPO-2	00:00:32
3	Pipeline 1	Extra Trees Regressor	2.070	None	00:00:01
4	Pipeline 2	Extra Trees Regressor	2.070	HPO-1	00:00:10
5	Pipeline 7	Decision Tree Regressor	2.714	HPO-1 FE	00:00:40
6	Pipeline 8	Decision Tree Regressor	2.714	HPO-1 FE HPO-2	00:00:06
7	Pipeline 5	Decision Tree Regressor	2.807	None	00:00:01
8	Pipeline 6	Decision Tree Regressor	2.807	HPO-1	00:00:01



IBM Watson Studio | Upgrade | Toyada Pattanaik's Account | TP

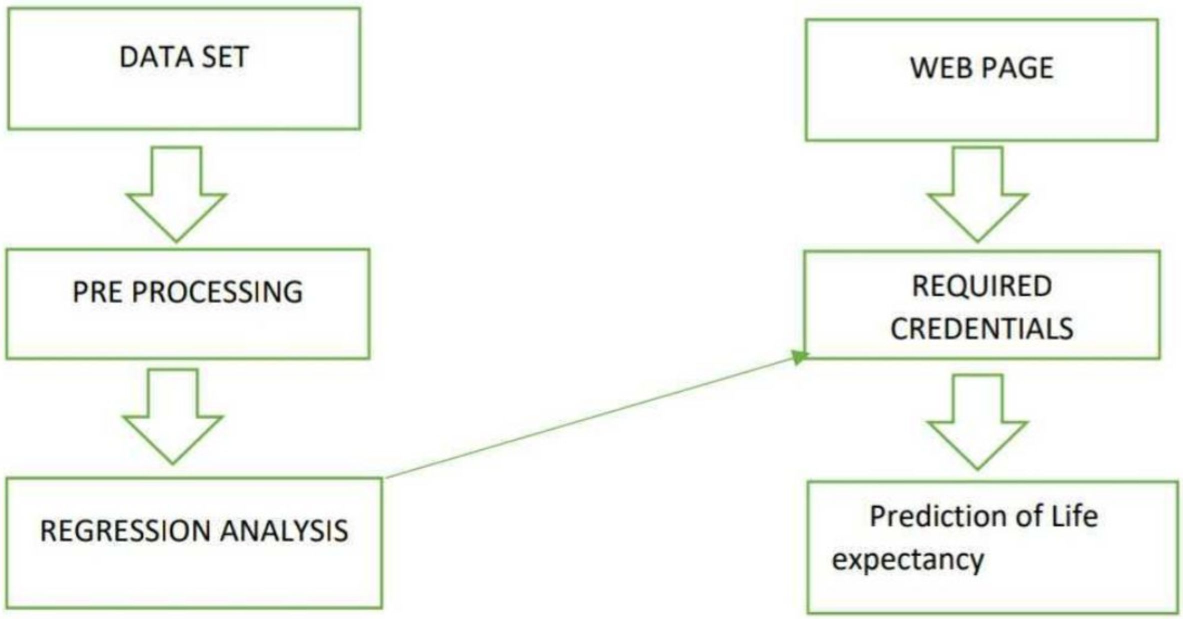
My projects / Life Expectancy Prediction / Life Expectancy Prediction

Experiment summary | Pipeline comparison | Rank by: Root mean squared err... | Score: Cross validation | Holdout

Pipeline leaderboard

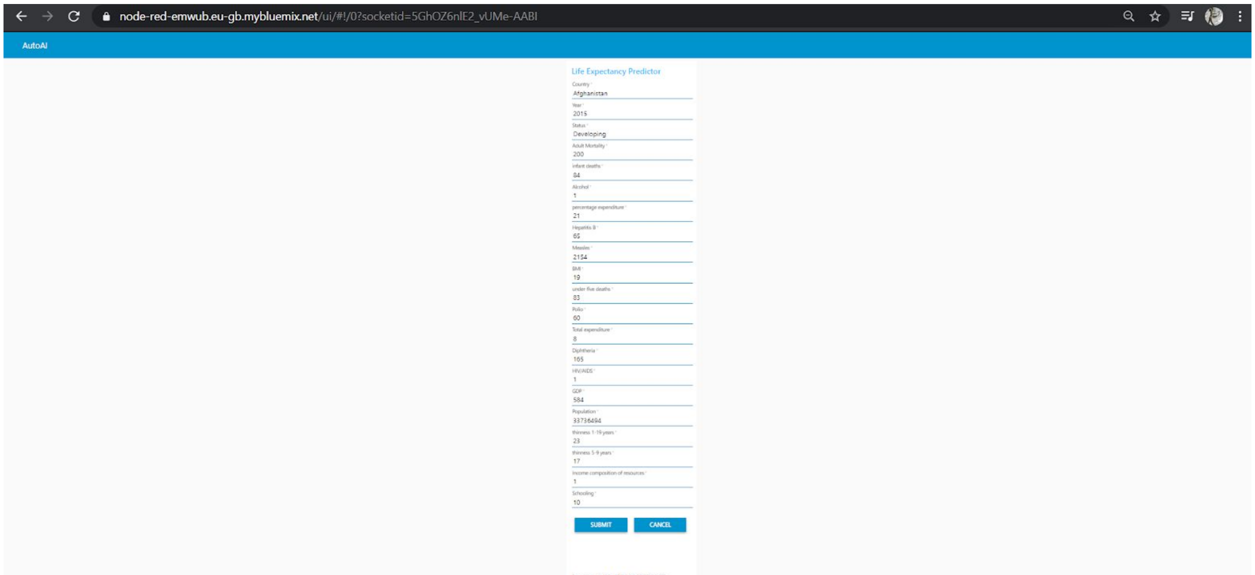
Rank	Name	Algorithm	RMSE (Optimized)	Explained va...	Mean absolu...	Mean square...	Mean square...	Median abso...	Root mean s...	R²
★ 1	Pipeline 3	Extra Trees Regressor	2.016	0.955	1.288	4.097	0.001	0.770	0.031	0.
2	Pipeline 4	Extra Trees Regressor	2.016	0.955	1.288	4.097	0.001	0.770	0.031	0.
3	Pipeline 1	Extra Trees Regressor	2.070	0.953	1.334	4.298	0.001	0.803	0.032	0.
4	Pipeline 2	Extra Trees Regressor	2.070	0.953	1.334	4.298	0.001	0.803	0.032	0.
5	Pipeline 7	Decision Tree Regressor	2.714	0.919	1.651	7.372	0.002	0.767	0.041	0.
6	Pipeline 8	Decision Tree Regressor	2.714	0.919	1.651	7.372	0.002	0.767	0.041	0.
7	Pipeline 5	Decision Tree Regressor	2.807	0.914	1.678	7.880	0.002	0.733	0.043	0.
8	Pipeline 6	Decision Tree Regressor	2.807	0.914	1.678	7.880	0.002	0.733	0.043	0.

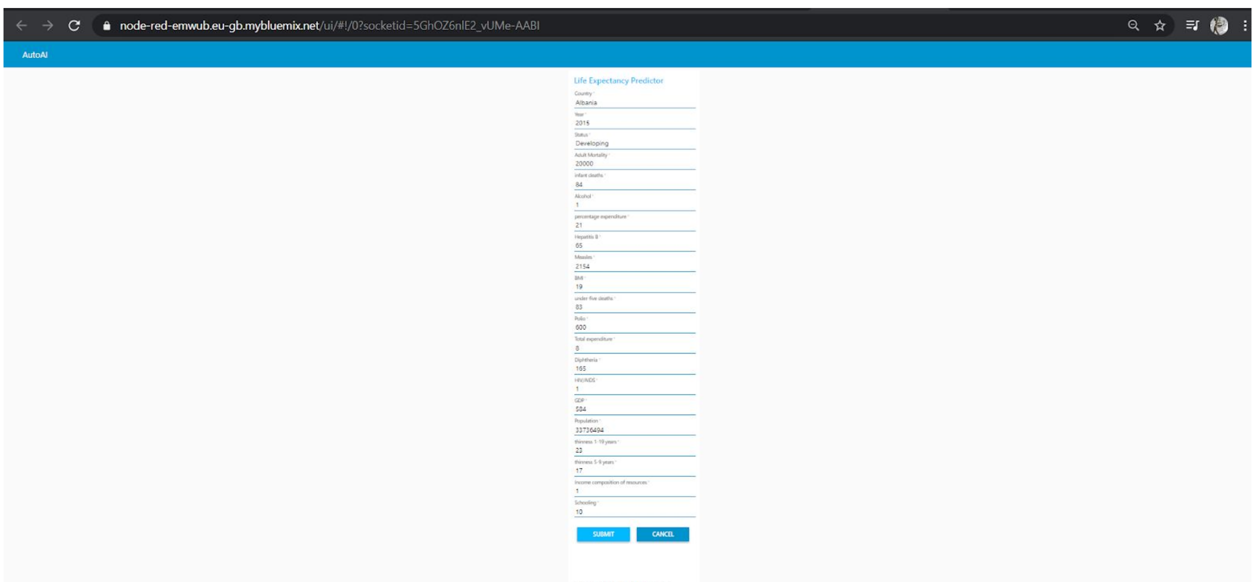
5 FLOWCHART



6 RESULT

At the end of the project, we had a simple web page with a form where users can enter random values for different variables affecting life expectancy and can visualize how all of them affect it differently. The Node-Red flow takes the inputted values and sends it to the machine learning model to process it. After processing, the result is sent back which is displayed below the form.





7 ADVANTAGES AND DISADVANTAGES

Advantages:

- It helps to provide some idea of where future data points will be. It enables a way to see into the future.

- It helps us to understand life expectancy's degree of dependency on various factors specified on the dataset.

Disadvantages:

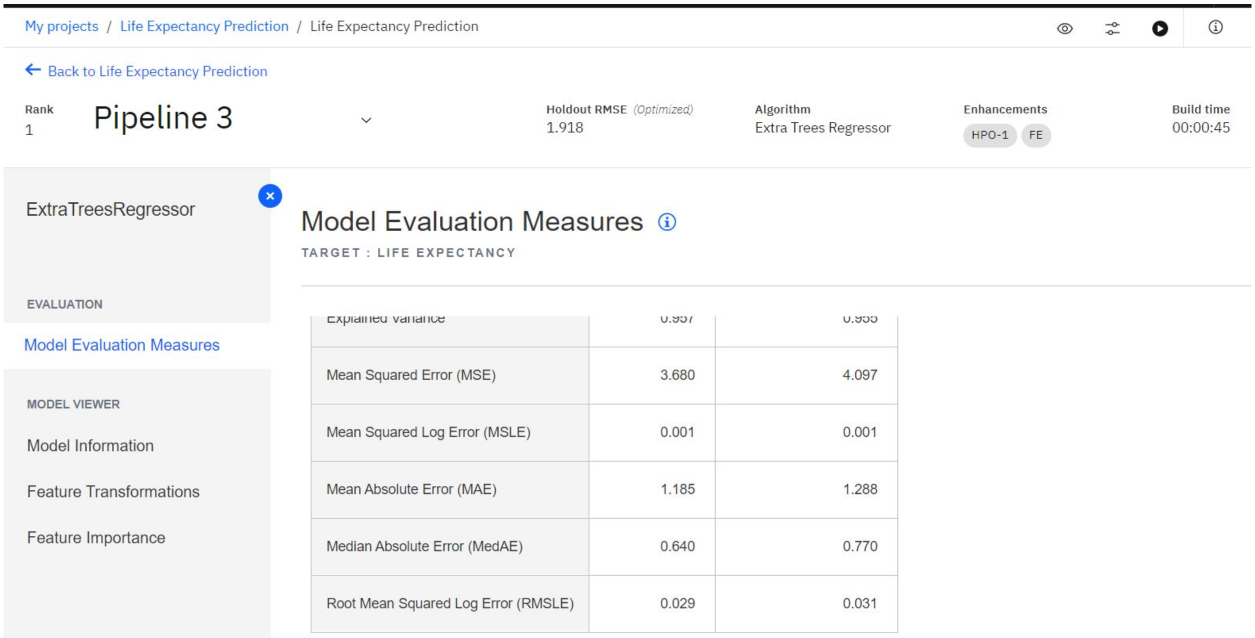
- It is limited to the linear relationship.
- Also, the regression solution is dense (because no regularization is applied).
- It is subject to overfitting.
- It is easily affected by outliers.

8 APPLICATIONS

- It can be used to predict life expectancy in future scenarios when factors will vary and dependency will change.
- It will help to direct focus towards resolving diseases which have a greater impact on life expectancy.
- Planning and management of data can be done more efficiently to produce better results.

9 CONCLUSION

After performing regression analysis using Auto AI, we conclude that Pipeline 3 is the most optimised method to predict life expectancy. The evaluation measures of our model is shown below,



10 FUTURE SCOPE

Machine Learning has huge potential to predict life expectancy in future. As we go forward, we can train our model on a massive scale of data which will in turn generate more accurate predictions. We can develop softwares which will evolve with time along with the ever changing factors.

In the end, we will have an advanced flexible system which will be able to predict the life expectancy of an individual or a country when specific values are provided.

11 BIBLIOGRAPHY

A. Source Code

Python Notebook of ML Model:

My projects / Life Expectancy Prediction / Life Expectancy Prediction - P3 s...

IBM AutoAI-SDK Auto-Generated Notebook v1.12.2

Note: Notebook code generated using AutoAI will execute successfully. If code is modified or reordered, there is no guarantee it will successfully execute. This pipeline is optimized for the original dataset. The pipeline may fail or produce sub-optimium results if used with different data. For different data, please consider returning to AutoAI Experiments to generate a new pipeline. Please read our documentation for more information:
Cloud Platform

Before modifying the pipeline or trying to re-fit the pipeline, consider:
The notebook converts dataframes to numpy arrays before fitting the pipeline
(a current restriction of the preprocessor pipeline). The known_values_list is passed by reference
and populated with categorical values during fit of the preprocessing pipeline. Delete its members before re-fitting.

Notebook content

This notebook contains steps and code to demonstrate AutoAI pipeline. This notebook introduces commands for getting data, pipeline model, model inspection and testing.

Some familiarity with Python is helpful. This notebook uses Python 3.

Notebook goals

- inspection of trained pipeline via graphical vizualization and source code preview.
- pipeline evaluation.
- pipeline deployment and webservice scoring

Contents

This notebook contains the following parts:

Setup

Before you use the sample code in this notebook, you must perform the following setup tasks:

- `watson-machine-learning-client` uninstallation of the old client
- `watson-machine-learning-client-V4` installation
- `autoai-libs` installation/upgrade
- `lightgbm` or `xgboost` installation/downgrade if they are needed

In [ ]:
```
!pip uninstall watson-machine-learning-client -y
```

In [ ]:
```
!pip install -U watson-machine-learning-client-V4
```

In [ ]:
```
!pip install -U autoai-libs
```

This cell contains input parameters provided to run the AutoAI experiment in Watson Studio and COS credentials required to retrieve AutoAI pipeline.

In [ ]:
```
from watson_machine_learning_client.helpers import DataConnection, S3Connection, S3Location

experiment_metadata = dict(
   prediction_type='regression',
   prediction_column='Life expectancy ',
   test_size=0.1,
   scoring='neg_root_mean_squared_error',
   max_number_of_estimators=2,
   training_data_reference = [DataConnection(
        connection=S3Connection(
            api_key='9FjOtY2tUlKJXtliO-bIlLM42y523WduG5Jjt2H5c2Mp',
            auth_endpoint='https://iam.bluemix.net/oidc/token/',
            endpoint_url='https://s3.eu-geo.objectstorage.softlayer.net'
        ),
            location=S3Location(
            bucket='lifeexpectancyprediction-donotdelete-pr-0btazi6eeksun6',
            path='Life Expectancy Data.csv'
        ))
    ],
    training_result_reference = DataConnection(
        connection=S3Connection(
            api_key='9FjOtY2tUlKJXtliO-bIlLM42y523WduG5Jjt2H5c2Mp',
            auth_endpoint='https://iam.bluemix.net/oidc/token/',
            endpoint_url='https://s3.eu-geo.objectstorage.softlayer.net'
        ),
        location=S3Location(
            bucket='lifeexpectancyprediction-donotdelete-pr-0btazi6eeksun6',
            path='auto_ml/e3dfea92-a184-4c75-8f75-902a88eeebba/wml_data/7886e992-c1a6-4907-b46e-25f05e703195/data/automl',
            model_location='auto_ml/e3dfea92-a184-4c75-8f75-902a88eeebba/wml_data/7886e992-c1a6-4907-b46e-25f05e703195/data/auto
ml/cognito_output/Pipeline1/model.pickle',
            training_status='auto_ml/e3dfea92-a184-4c75-8f75-902a88eeebba/wml_data/7886e992-c1a6-4907-b46e-25f05e703195/training
-status.json'
        )
    ))

pipeline_name='Pipeline_3'
```


Get historical optimizer instance

The next cell contains code for retrieving fitted optimizer.

In [ ]:
```
from watson_machine_learning_client.experiment import AutoAI

optimizer = AutoAI().runs.get_optimizer(metadata=experiment_metadata)
```

Get pipeline model

The following cell loads selected AutoAI pipeline model. If you want to get pure scikit-learn pipeline specify `as_type='sklearn'` parameter. By default enriched scikit-learn pipeline is returned `as_type='lale'`.

In [ ]:
```
pipeline_model = optimizer.get_pipeline(pipeline_name=pipeline_name)
```

Preview pipeline model as python code

In the next cell, downloaded pipeline model could be previewed as a python code.
You will be able to see what exact steps are involved in model creation.

In [ ]:
```
pipeline_model.pretty_print(combinators=False, ipython_display=True)
```

Visualize pipeline model

Preview pipeline model stages as graph. Each node's name links to detailed description of the stage.

In [ ]:
```
pipeline_model.visualize()
```

Read training and holdout data

Retrieve training dataset from AutoAI experiment as pandas DataFrame.

In [ ]:
```
training_df, holdout_df = optimizer.get_data_connections()[0].read(with_holdout_split=True)

train_X = training_df.drop([experiment_metadata['prediction_column']], axis=1).values
train_y = training_df[experiment_metadata['prediction_column']].values

test_X = holdout_df.drop([experiment_metadata['prediction_column']], axis=1).values
y_true = holdout_df[experiment_metadata['prediction_column']].values
```

Test pipeline model locally

Note: you can chose the metric to evaluate the model by your own, this example contains only a basic scenario.

In [ ]:
```
from sklearn.metrics import r2_score

predictions = pipeline_model.predict(test_X)
score = r2_score(y_true=y_true, y_pred=predictions)
print('r2_score: ', score)
```

Pipeline refinery and testing (optional)

In this section you will learn how to refine and retrain the best pipeline returned by AutoAI. It can be performed by:

- modifying pipeline definition source code
- using lale library for semi-automated data science

Note: In order to run this section change following cells to 'code' cell.

My projects / Life Expectancy Prediction / Life Expectancy Prediction - P3 s...

Deploy and Score

In this section you will learn how to deploy and score pipeline model as webservice using WML instance.

Connect to WML client in order to create deployment

Action: Next you will need credentials for Watson Machine Learning and training run_id:

- go to Cloud catalog resources list
- click on Services and chose Machine Learning service. Once you are there
- click the **Service Credentials** link on the left side of the screen
- click to expand specific credentials name.
- copy and paste your WML credentials into the cell below

Take in mind that WML Service instance should be the same as used to generate this notebook.

In [ ]:
```
wml_credentials = {
  "apikey": "",
  "iam_apikey_description": "",
  "iam_apikey_name": "",
  "iam_role_crn": "r",
  "iam_serviceid_crn": "",
  "instance_id": "",
  "url": ""
}
```

My projects / Life Expectancy Prediction / Life Expectancy Prediction - P3 s...

Create deployment

Action: If you want to deploy refined pipeline please change the `pipeline_model` to `new_pipeline`.

If you prefer you can also change the `deployment_name`.

In [ ]:
```
from watson_machine_learning_client.deployment import WebService

service = WebService(wml_credentials)

service.create(
    model=pipeline_model,
    metadata=experiment_metadata,
    deployment_name=f'{pipeline_name}_webservice'
)
```

Deployment object could be printed to show basic information:

In [ ]:
```
print(service)
```

To be able to show all available information about deployment use `.get_params()` method:

In [ ]:
```
service.get_params()
```

Score webservice

You can make scoring request by calling `score()` on deployed pipeline.

In [ ]:
```
predictions = service.score(payload=holdout_df.drop([experiment_metadata['prediction_column']], axis=1).iloc[:10])
predictions
```

JSON Script of Node-Red flow:

[{"id":"7f4dbbe6.6c04f4","type":"tab","label":"Life Expectancy Prediction using Auto
AI","disabled":false,"info":""},{"id":"10c3b9f3.5185a6","type":"ui_tab","z":"","name":"Home","icon":"dashboard",
"disabled":false,"hidden":false},{"id":"50805e17.a9c9d","type":"ui_group","z":"","name":"Life Expectancy
Predictor","tab":"10c3b9f3.5185a6","order":2,"disp":true,"width":"6","collapse":false},{"id":"b05e5433.ba3db8","t
ype":"ui_base","theme":{"name":"theme-light","lightTheme":{"default":"#0094CE","baseColor":"#0094CE","base
Font":"-apple-system,BlinkMacSystemFont,Segoe UI,Roboto,Oxygen-Sans,Ubuntu,Cantarell,Helvetica
Neue,sans-serif","edited":true,"reset":false},"darkTheme":{"default":"#097479","baseColor":"#097479","baseFont":
"-apple-system,BlinkMacSystemFont,Segoe UI,Roboto,Oxygen-Sans,Ubuntu,Cantarell,Helvetica
Neue,sans-serif","edited":false},"customTheme":{"name":"Untitled Theme
1","default":"#4B7930","baseColor":"#4B7930","baseFont":"-apple-system,BlinkMacSystemFont,Segoe
UI,Roboto,Oxygen-Sans,Ubuntu,Cantarell,Helvetica
Neue,sans-serif"},"themeState":{"base-color":{"default":"#0094CE","value":"#0094CE","edited":false},"page-titleb
ar-backgroundColor":{"value":"#0094CE","edited":false},"page-backgroundColor":{"value":"#fafafa","edited":fals
e},"page-sidebar-backgroundColor":{"value":"#ffffff","edited":false},"group-textColor":{"value":"#1bbfff","edited"
:false},"group-borderColor":{"value":"#ffffff","edited":false},"group-backgroundColor":{"value":"#ffffff","edited":
false},"widget-textColor":{"value":"#111111","edited":false},"widget-backgroundColor":{"value":"#0094ce","edite
d":false},"widget-borderColor":{"value":"#ffffff","edited":false},"base-font":{"value":"-apple-system,BlinkMacSys
temFont,Segoe UI,Roboto,Oxygen-Sans,Ubuntu,Cantarell,Helvetica
Neue,sans-serif"}},"angularTheme":{"primary":"indigo","accents":"blue","warn":"red","background":"grey"}},"site
":{"name":"Node-RED
Dashboard","hideToolbar":"false","allowSwipe":"false","lockMenu":"false","allowTempTheme":"true","dateFormat
":"DD/MM/YYYY","sizes":{"sx":48,"sy":48,"gx":6,"gy":6,"cx":6,"cy":6,"px":0,"py":0}}},{"id":"d6500cd.4fb42f",
"type":"ui_group","z":"","name":"Life Expectancy
Predictor","tab":"5475484d.671f98","order":2,"disp":true,"width":"6","collapse":false},{"id":"5475484d.671f98","t
ype":"ui_tab","z":"","name":"AutoAI","icon":"dashboard","order":2,"disabled":false,"hidden":false},{"id":"1d648c
89.cc6c93","type":"function","z":"7f4dbbe6.6c04f4","name":"Pre
token","func":"global.set(\"co\",msg.payload.co)\nglobal.set(\"ye\",msg.payload.ye)\nglobal.set(\"st\",msg.payload.s
t)\nglobal.set(\"adult\",msg.payload.adult)\nglobal.set(\"infant\",msg.payload.infant)\nglobal.set(\"alc\",msg.payload
.alc)\nglobal.set(\"per_exp\",msg.payload.per_exp)\nglobal.set(\"hepB\",msg.payload.hepB)\nglobal.set(\"meas\",ms
g.payload.meas)\nglobal.set(\"bmi\",msg.payload.bmi)\nglobal.set(\"under\",msg.payload.under)\nglobal.set(\"polio\
",msg.payload.polio)\nglobal.set(\"tot_exp\",msg.payload.tot_exp)\nglobal.set(\"diph\",msg.payload.diph)\nglobal.se
t(\"hiv\",msg.payload.hiv)\nglobal.set(\"gdp\",msg.payload.gdp)\nglobal.set(\"popu\",msg.payload.popu)\nglobal.set
(\"thin1\",msg.payload.thin1)\nglobal.set(\"thin5\",msg.payload.thin5)\nglobal.set(\"income\",msg.payload.income)\
nglobal.set(\"school\",msg.payload.school)\nvar
apikey=\"KjaZGdmAHJl5ECDEa3br-Zw81bTBq1o1vZx3MBeKhu8I\";\nmsg.headers={\"content-type\":\"applicati
on/x-www-form-urlencoded\"}\nmsg.payload={\"grant_type\":\"urn:ibm:params:oauth:grant-type:apikey\",\"apikey\
":apikey}\nreturn
msg;","outputs":1,"noerr":0,"x":260,"y":280,"wires":[["a89041c3.5ab2d"]]},{"id":"a89041c3.5ab2d","type":"http
request","z":"7f4dbbe6.6c04f4","name":"","method":"POST","ret":"obj","paytoqs":false,"url":"https://iam.cloud.ib
m.com/identity/token","tls":"","persist":false,"proxy":"","authType":"","x":430,"y":280,"wires":[["1d9b245f.737dec
","b599d5fc.31f438"]]},{"id":"9a43f05f.d48b9","type":"debug","z":"7f4dbbe6.6c04f4","name":"","active":true,"tosi
debar":true,"console":false,"tostatus":false,"complete":"payload","targetType":"msg","x":890,"y":60,"wires":[]},{"i
d":"1d9b245f.737dec","type":"function","z":"7f4dbbe6.6c04f4","name":"sendto Endpoint","func":"var co =
global.get('co')\nvar ye = global.get('ye')\nvar st = global.get('st')\nvar adult = global.get('adult')\nvar infant =
global.get('infant')\nvar alc = global.get('alc')\nvar per_exp = global.get('per_exp')\nvar hepB =

global.get('hepB')\nvar meas = global.get('meas')\nvar bmi = global.get('bmi')\nvar under = global.get('under')\nvar polio = global.get('polio')\nvar tot_exp = global.get('tot_exp')\nvar diph = global.get('diph')\nvar hiv = global.get('hiv')\nvar gdp = global.get('gdp')\nvar popu = global.get('popu')\nvar thin1 = global.get('thin1')\nvar thin5 = global.get('thin5')\nvar income = global.get('income')\nvar school = global.get('school')\nvar token=msg.payload.access_token\nvar
instance_id=\"919d0abc-349b-4648-a6fb-ee98423bc04e\"\nmsg.headers={'Content-Type':
'application/json',\"Authorization\":\"Bearer
\"+token,\"ML-Instance-ID\":instance_id}\nmsg.payload={\"input_data\": [{\"fields\": [\"Country\", \"Year\", \"Status\", \"Adult Mortality\", \"infant deaths\", \"Alcohol\", \"percentage expenditure\", \"Hepatitis B\", \"Measles \", \" BMI \", \"under-five deaths \", \"Polio\", \"Total expenditure\", \"Diphtheria \", \" HIV/AIDS\", \"GDP\", \"Population\", \" thinness 1-19 years\", \" thinness 5-9 years\", \"Income composition of resources\", \"Schooling\"], \"values\":
[[co,ye,st,adult,infant,alc,per_exp,hepB,meas,bmi,under,polio,tot_exp,diph,hiv,gdp,popu,thin1,thin5,income,school]]}]}\nreturn
msg;","outputs":1,"noerr":0,"x":540,"y":200,"wires":[["c9ac629d.203e8"]]},{"id":"c9ac629d.203e8","type":"http request","z":"7f4dbbe6.6c04f4","name":"","method":"POST","ret":"obj","paytoqs":false,"url":"https://eu-gb.ml.cloud.ibm.com/v4/deployments/ee8233e7-b357-4733-9e77-fcc2394a3492/predictions","tls":"","persist":false,"proxy":"","authType":"","x":710,"y":300,"wires":[["68b090c1.1a8b5","bd68dc9a.8971c"]]},{"id":"54923a3.f6d56c4","type":"ui_form","z":"7f4dbbe6.6c04f4","name":"","label":"","group":"d6500cd.4fb42f","order":1,"width":0,"height":0,"options":[{"label":"Country","value":"co","type":"text","required":true,"rows":null},{"label":"Year","value":"ye","type":"number","required":true,"rows":null},{"label":"Status","value":"st","type":"text","required":true,"rows":null},{"label":"Adult Mortality","value":"adult","type":"number","required":true,"rows":null},{"label":"infant deaths","value":"infant","type":"number","required":true,"rows":null},{"label":"Alcohol","value":"alc","type":"number","required":true,"rows":null},{"label":"percentage
expenditure","value":"per_exp","type":"number","required":true,"rows":null},{"label":"Hepatitis
B","value":"hepB","type":"number","required":true,"rows":null},{"label":"Measles","value":"meas","type":"number","required":true,"rows":null},{"label":"BMI","value":"bmi","type":"number","required":true,"rows":null},{"label":"under-five
deaths","value":"under","type":"number","required":true,"rows":null},{"label":"Polio","value":"polio","type":"number","required":true,"rows":null},{"label":"Total
expenditure","value":"tot_exp","type":"number","required":true,"rows":null},{"label":"Diphtheria
","value":"diph","type":"number","required":true,"rows":null},{"label":"
HIV/AIDS","value":"hiv","type":"number","required":true,"rows":null},{"label":"GDP","value":"gdp","type":"number","required":true,"rows":null},{"label":"Population","value":"popu","type":"number","required":true,"rows":null},{"label":"thinness 1-19 years","value":"thin1","type":"number","required":true,"rows":null},{"label":"thinness 5-9 years","value":"thin5","type":"number","required":true,"rows":null},{"label":"Income composition of resources","value":"income","type":"number","required":true,"rows":null},{"label":"Schooling","value":"school","type":"number","required":true,"rows":null}],"formValue":{"co":"","ye":"","st":"","adult":"","infant":"","alc":"","per_exp":"","hepB":"","meas":"","bmi":"","under":"","polio":"","tot_exp":"","diph":"","hiv":"","gdp":"","popu":"","thin1":"","thin5":"","income":"","school":""},"payload":"","submit":"submit","cancel":"cancel","topic":"","x":90,"y":320,"wires":[["1d648c89.cc6c93","83cdfc4b.45159"]]},{"id":"f502ea73.da9788","type":"ui_text","z":"7f4dbbe6.6c04f4","group":"d6500cd.4fb42f","order":2,"width":0,"height":0,"name":"","label":"Prediction:","format":"{{msg.payload}}","layout":"row-left","x":830,"y":140,"wires":[]},{"id":"bd68dc9a.8971c","type":"function","z":"7f4dbbe6.6c04f4","name":"","func":"msg.payload=msg.payload.predictions[0].values[0][0]\nreturn
msg;\n","outputs":1,"noerr":0,"x":670,"y":80,"wires":[["9a43f05f.d48b9","f502ea73.da9788"]]},{"id":"b599d5fc.31f438","type":"debug","z":"7f4dbbe6.6c04f4","name":"","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"payload","targetType":"msg","x":550,"y":380,"wires":[]},{"id":"68b090c1.1a8b5","type":"debug","z":"7f4dbbe6.6c04f4","name":"","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"payload","ta

rgetType":"msg","x":910,"y":380,"wires":[]},{"id":"ce194e26.92b1d","type":"debug","z":"7f4dbbe6.6c04f4","name":"","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"payload","targetType":"msg","x":370,"y":460,"wires":[]},{"id":"83cdfc4b.45159","type":"function","z":"7f4dbbe6.6c04f4","name":"","func":"\nreturn msg;","outputs":1,"noerr":0,"x":250,"y":380,"wires":[["ce194e26.92b1d"]]}]