Interpretable classification model(Glass box model)

In [1]:
```
import pandas as pd
from sklearn.model_selection import train_test_split
```

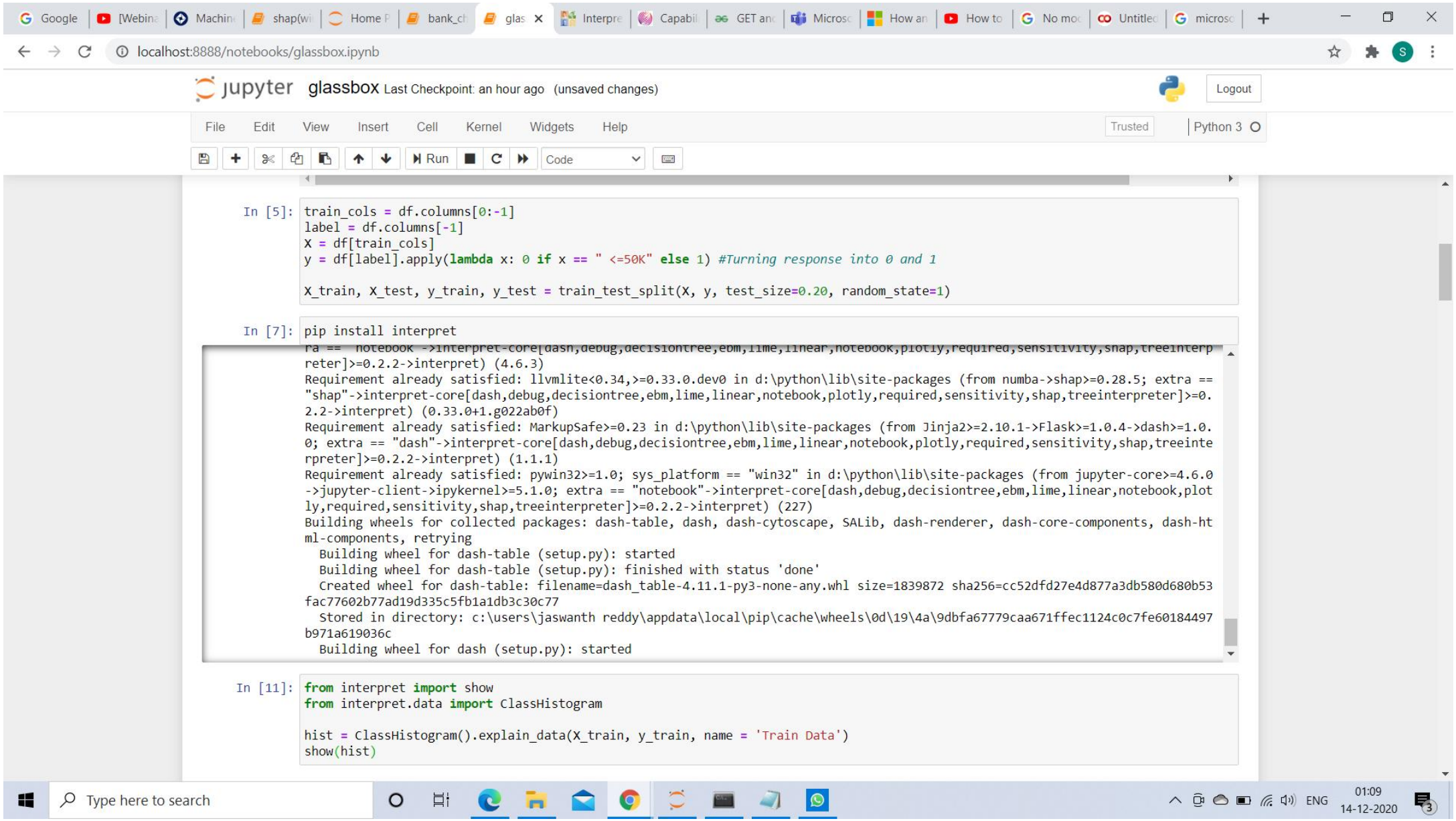
In [2]:
```
df = pd.read_csv(r"C:\Users\Jaswanth Reddy\Downloads\adult.data",
    header=None)
df.columns = [
    "Age", "WorkClass", "fnlwgt", "Education", "EducationNum",
    "MaritalStatus", "Occupation", "Relationship", "Race", "Gender",
    "CapitalGain", "CapitalLoss", "HoursPerWeek", "NativeCountry", "Income"
]
```

In [3]:
```
df.head()
```

Out[3]:

	Age	WorkClass	fnlwgt	Education	EducationNum	MaritalStatus	Occupation	Relationship	Race	Gender	CapitalGain	CapitalLoss	HoursPerWeek	Nativ
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male	2174	0	40	Uni
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male	0	0	13	Uni
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male	0	0	40	Uni
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male	0	0	40	Uni
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female	0	0	40	

In [5]:
```
train_cols = df.columns[0:-1]
```


localhost:8888/notebooks/glassbox.ipynb
jupyter glassbox Last Checkpoint: an hour ago (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
In [5]:
train_cols = df.columns[0:-1]
label = df.columns[-1]
X = df[train_cols]
y = df[label].apply(lambda x: 0 if x == " <=50K" else 1) #Turning response into 0 and 1
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state=1)
In [7]:
pip install interpret
ra == "notebook" ->interpret-core[dash,debug,decisiontree,ebm,lime,linear,notebook,plotly,required,sensitivity,shap,treeinterpreter]>=0.2.2->interpret) (4.6.3)
Requirement already satisfied: llvmlite<0.34,>=0.33.0.dev0 in d:\python\lib\site-packages (from numba->shap>=0.28.5; extra == "shap"->interpret-core[dash,debug,decisiontree,ebm,lime,linear,notebook,plotly,required,sensitivity,shap,treeinterpreter]>=0.2.2->interpret) (0.33.0+1.g022ab0f)
Requirement already satisfied: MarkupSafe>=0.23 in d:\python\lib\site-packages (from Jinja2>=2.10.1->Flask>=1.0.4->dash>=1.0.0; extra == "dash"->interpret-core[dash,debug,decisiontree,ebm,lime,linear,notebook,plotly,required,sensitivity,shap,treeinterpreter]>=0.2.2->interpret) (1.1.1)
Requirement already satisfied: pywin32>=1.0; sys_platform == "win32" in d:\python\lib\site-packages (from jupyter-core>=4.6.0->jupyter-client->ipykernel>=5.1.0; extra == "notebook"->interpret-core[dash,debug,decisiontree,ebm,lime,linear,notebook,plotly,required,sensitivity,shap,treeinterpreter]>=0.2.2->interpret) (227)
Building wheels for collected packages: dash-table, dash, dash-cytoscape, SALib, dash-renderer, dash-core-components, dash-html-components, retrying
Building wheel for dash-table (setup.py): started
Building wheel for dash-table (setup.py): finished with status 'done'
Created wheel for dash-table: filename=dash_table-4.11.1-py3-none-any.whl size=1839872 sha256=cc52dfd27e4d877a3db580d680b53fac77602b77ad19d335c5fb1a1db3c30c77
Stored in directory: c:\users\jaswanth reddy\appdata\local\pip\cache\wheels\0d\19\4a\9dbfa67779caa671ffec1124c0c7fe60184497b971a619036c
Building wheel for dash (setup.py): started
In [11]:
from interpret import show
from interpret.data import ClassHistogram
hist = ClassHistogram().explain_data(X_train, y_train, name = 'Train Data')
show(hist)
Type here to search
ENG
01:09
14-12-2020

Google | [Webina | Machin | shap(wi | Home P | bank_ch | glas | Interpre | Capabil | GET anc | Microsc | How an | How to | No moc | Untitlec | microsc

localhost:8888/notebooks/glassbox.ipynb

jupyter glassbox Last Checkpoint: an hour ago (unsaved changes)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3

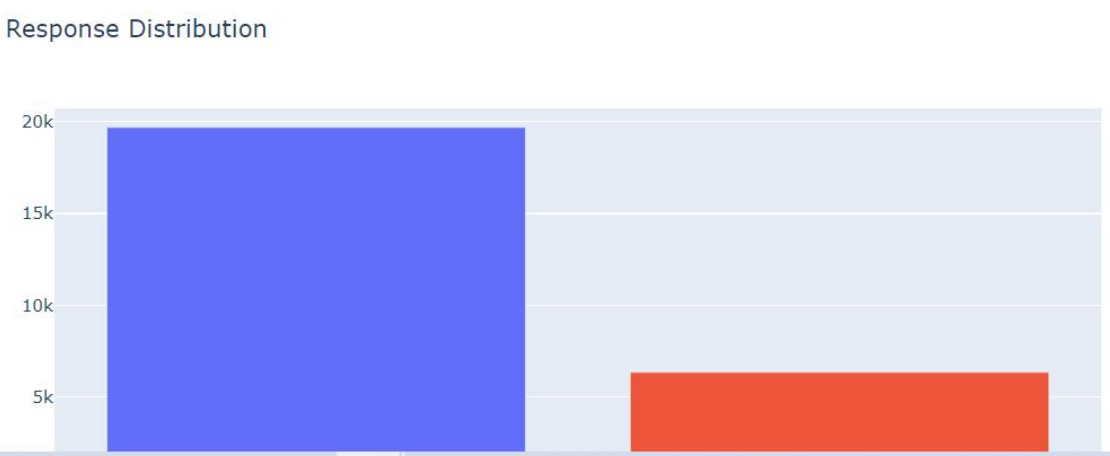
Run Code

```
show(hist)
```

Select Component to Graph

Summary

Train Data (Overall)



Type here to search

ENG 01:10 14-12-2020

Google | [Webina | Machin | shap(wi | Home P | bank_ch | glas | Interpre | Capabil | GET anc | Microsc | How an | How to | No mod | Untitlec | microsc

localhost:8888/notebooks/glassbox.ipynb

jupyter glassbox Last Checkpoint: an hour ago (unsaved changes)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3

Run | Code

In [12]:
```
# Training the model
from interpret.glassbox import ExplainableBoostingClassifier, LogisticRegression, ClassificationTree, DecisionListClassifier

ebm = ExplainableBoostingClassifier(random_state=1)
ebm.fit(X_train, y_train)
```

Out[12]:
```
ExplainableBoostingClassifier(feature_names=['Age', 'WorkClass', 'fnlwgt',
                                             'Education', 'EducationNum',
                                             'MaritalStatus', 'Occupation',
                                             'Relationship', 'Race', 'Gender',
                                             'CapitalGain', 'CapitalLoss',
                                             'HoursPerWeek', 'NativeCountry'],
                              feature_types=['continuous', 'categorical',
                                             'continuous', 'categorical',
                                             'continuous', 'categorical',
                                             'categorical', 'categorical',
                                             'categorical', 'categorical',
                                             'continuous', 'continuous',
                                             'continuous', 'categorical'],
                              random_state=1)
```

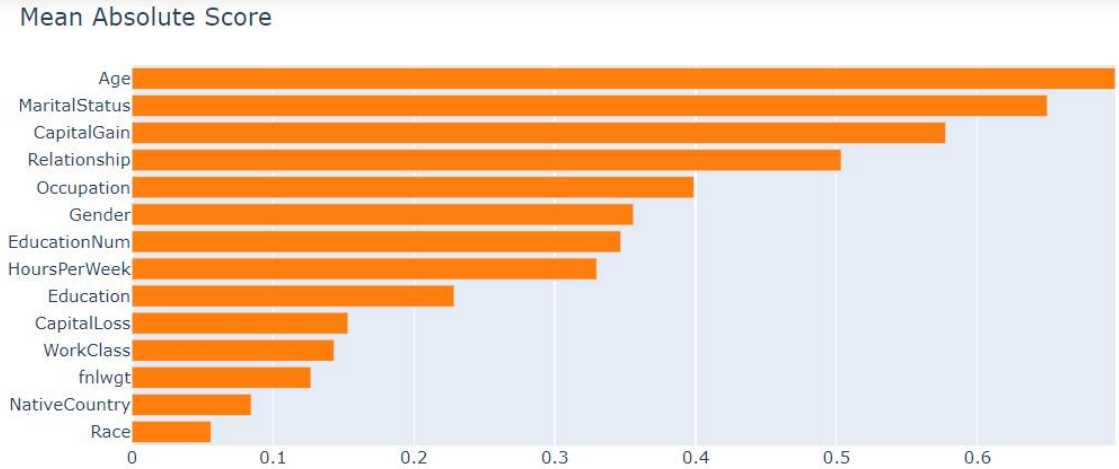
In [13]:
```
# Global Explanations: What the model learned overall
ebm_global = ebm.explain_global(name='EBM')
show(ebm_global)
```

Select Component to Graph

Summary

Type here to search

ENG 01:10 14-12-2020



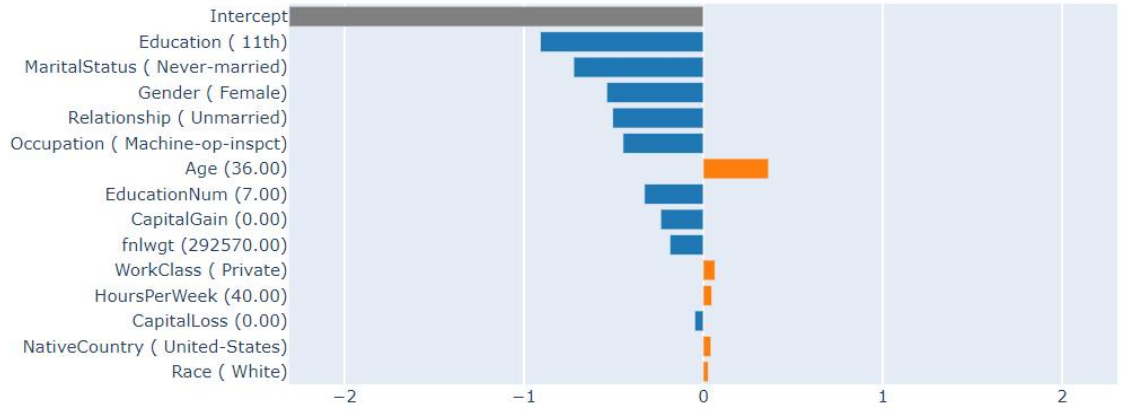
In [15]:
```
# Local Explanations: How an individual prediction was made
ebm_local = ebm.explain_local(X_test[:5], y_test[:5], name='EBM')
show(ebm_local)
```

Select Component to Graph

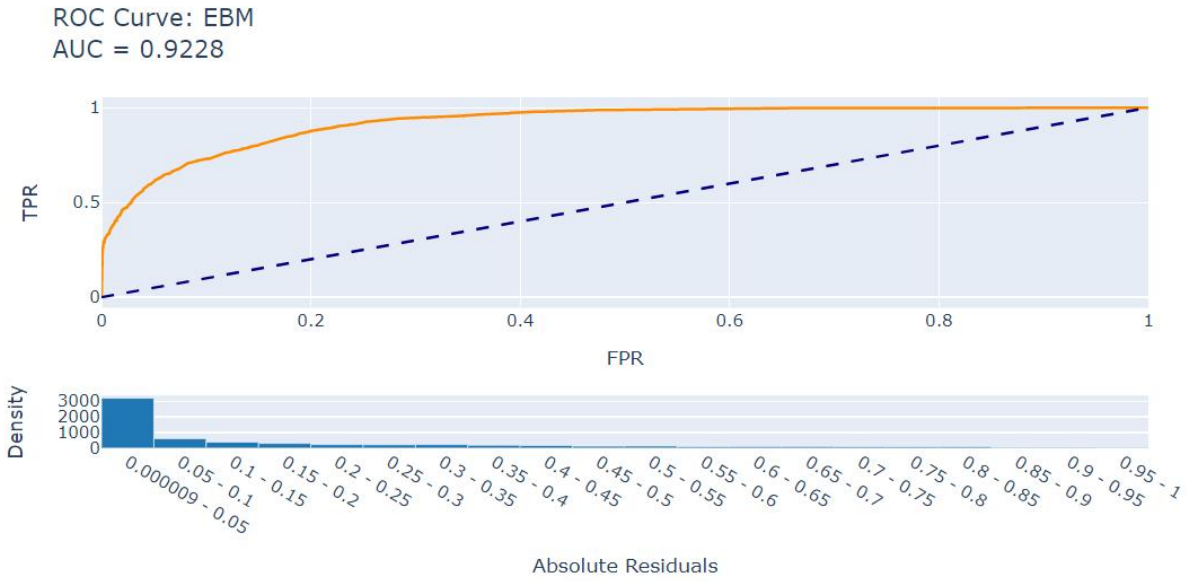
jupyter glassbox Last Checkpoint: an hour ago (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code



In [16]:
```
# Evaluating performance
from interpret.perf import ROC

ebm_perf = ROC(ebm.predict_proba).explain_perf(X_test, y_test, name='EBM')
show(ebm_perf)
```



localhost:8888/notebooks/glassbox.ipynb

jupyter glassbox Last Checkpoint: an hour ago (unsaved changes)

Logout

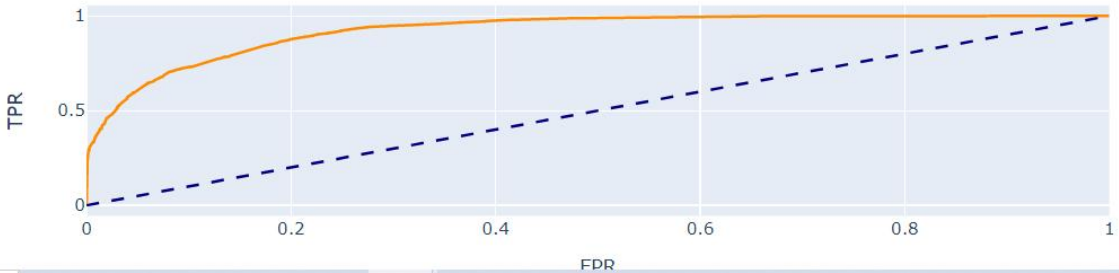
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code

In [18]:
```
# Other models
from interpret.glassbox import LogisticRegression, ClassificationTree
seed=1
# We have to transform categorical variables to use Logistic Regression and Decision Tree
X_enc = pd.get_dummies(X, prefix_sep='.')
feature_names = list(X_enc.columns)
X_train_enc, X_test_enc, y_train, y_test = train_test_split(X_enc, y, test_size=0.20, random_state=seed)

lr = LogisticRegression(random_state=seed, feature_names=feature_names, penalty='l1', solver='liblinear')
lr.fit(X_train_enc, y_train)

tree = ClassificationTree()
tree.fit(X_train_enc, y_train)
```

Out[18]: <interpret.glassbox.decisiontree.ClassificationTree at 0x2533929ac10>

In [19]:
```
# Performance of all models
lr_perf = ROC(lr.predict_proba).explain_perf(X_test_enc, y_test, name='Logistic Regression')
tree_perf = ROC(tree.predict_proba).explain_perf(X_test_enc, y_test, name='Classification Tree')

show(lr_perf)
show(tree_perf)
show(ebm_perf)
```



Type here to search

ENG 01:10 14-12-2020

Google | [Webina | Machin | shap(wi | Home P | bank_ch | glas | Interpre | Capabil | GET anc | Microsc | How an | How to | No mod | Untitled | microsc

localhost:8888/notebooks/glassbox.ipynb

jupyter glassbox Last Checkpoint: an hour ago (unsaved changes)

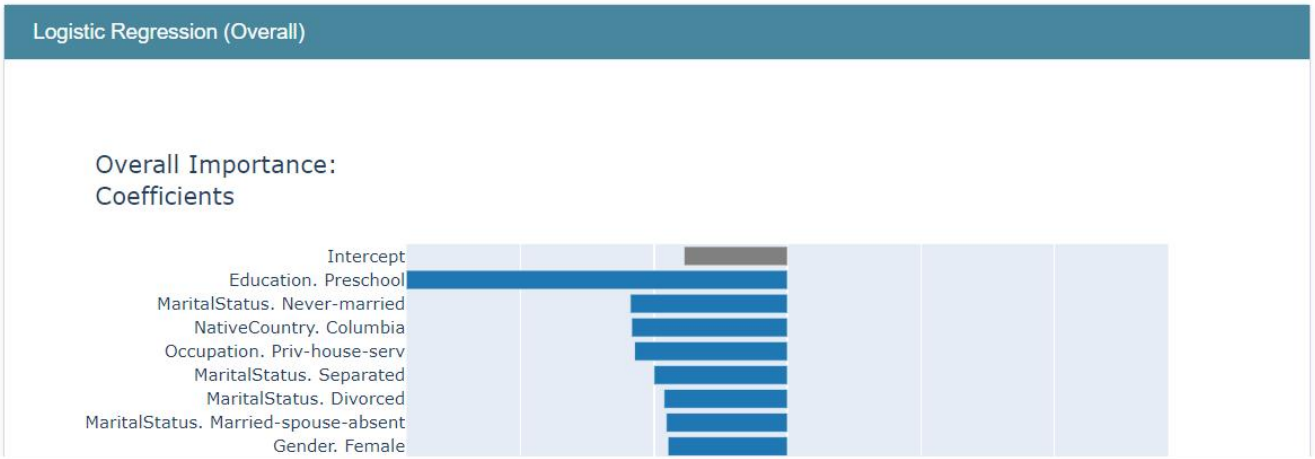
Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3

Run | Code

In [20]:
```
# Gloabal explanation
lr_global = lr.explain_global(name='Logistic Regression')
tree_global = tree.explain_global(name='Classification Tree')

show(lr_global)
show(tree_global)
show(ebm_global)
```



In [ ]:

Type here to search

ENG 01:10 14-12-2020

Google | [Webina | Machin | shap(wi | Home P | bank_ch | glas | Interpre | Capabil | GET anc | Microsc | How an | How to | No moc | Untitlec | microsc

localhost:8888/notebooks/glassbox.ipynb

jupyter glassbox Last Checkpoint: an hour ago (unsaved changes)

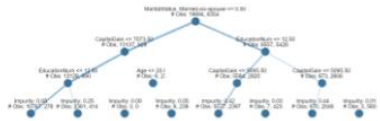
Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3

Run | Code

In [20]:
```
# Gloabal explanation
lr_global = lr.explain_global(name='Logistic Regression')
tree_global = tree.explain_global(name='Classification Tree')

show(lr_global)
show(tree_global)
show(ebm_global)
```



In [ ]:

Type here to search

ENG 01:10 14-12-2020

Google | [Webina | Machin | shap(wi | Home P | bank_ch | glas | Interpre | Capabil | GET anc | Microsc | How an | How to | No moc | Untitlec | microsc

localhost:8888/notebooks/glassbox.ipynb

jupyter glassbox Last Checkpoint: an hour ago (unsaved changes)

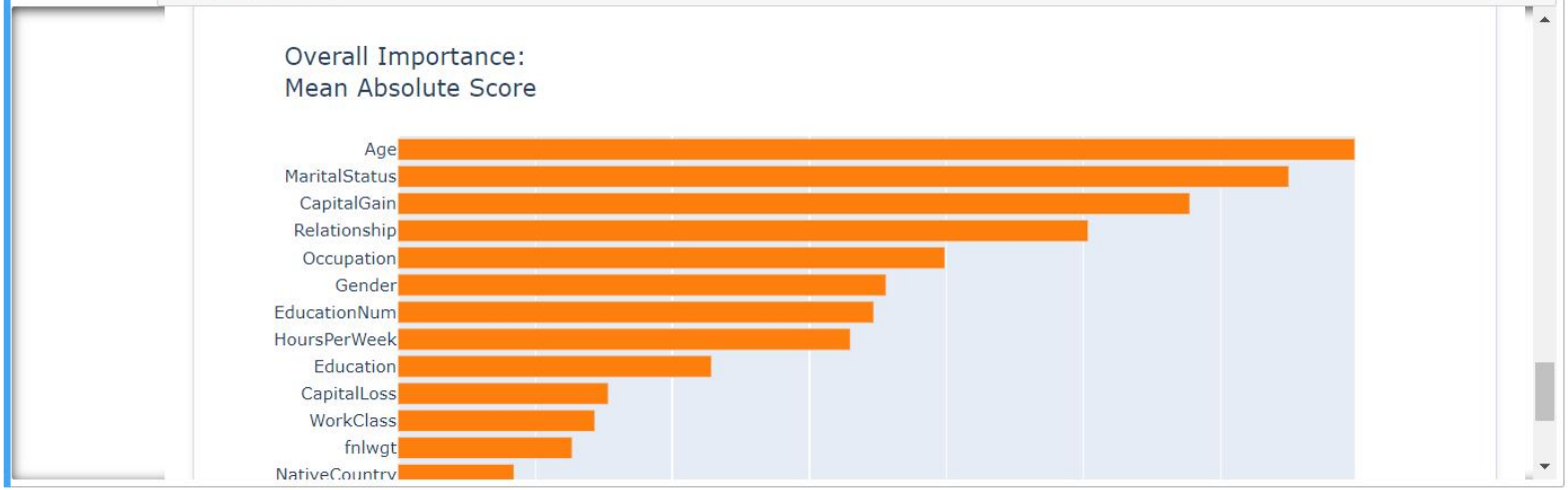
Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3

Run | Code

In [20]:
```
# Gloabal explanation
lr_global = lr.explain_global(name='Logistic Regression')
tree_global = tree.explain_global(name='Classification Tree')

show(lr_global)
show(tree_global)
show(ebm_global)
```



In [ ]:

Type here to search

ENG 01:11 14-12-2020