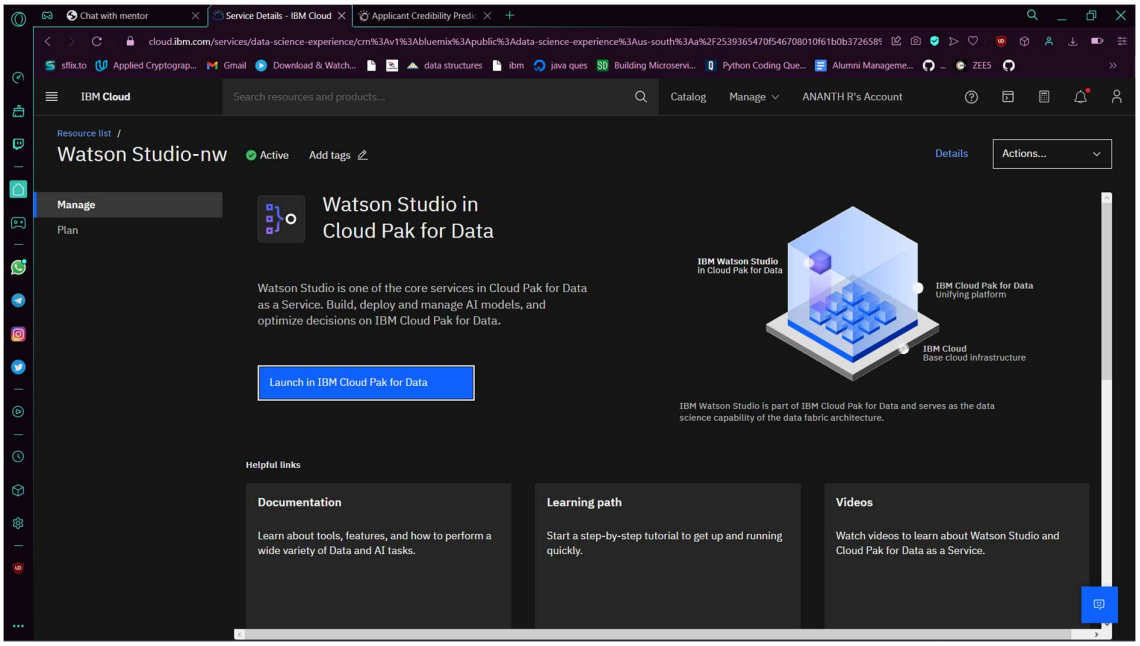
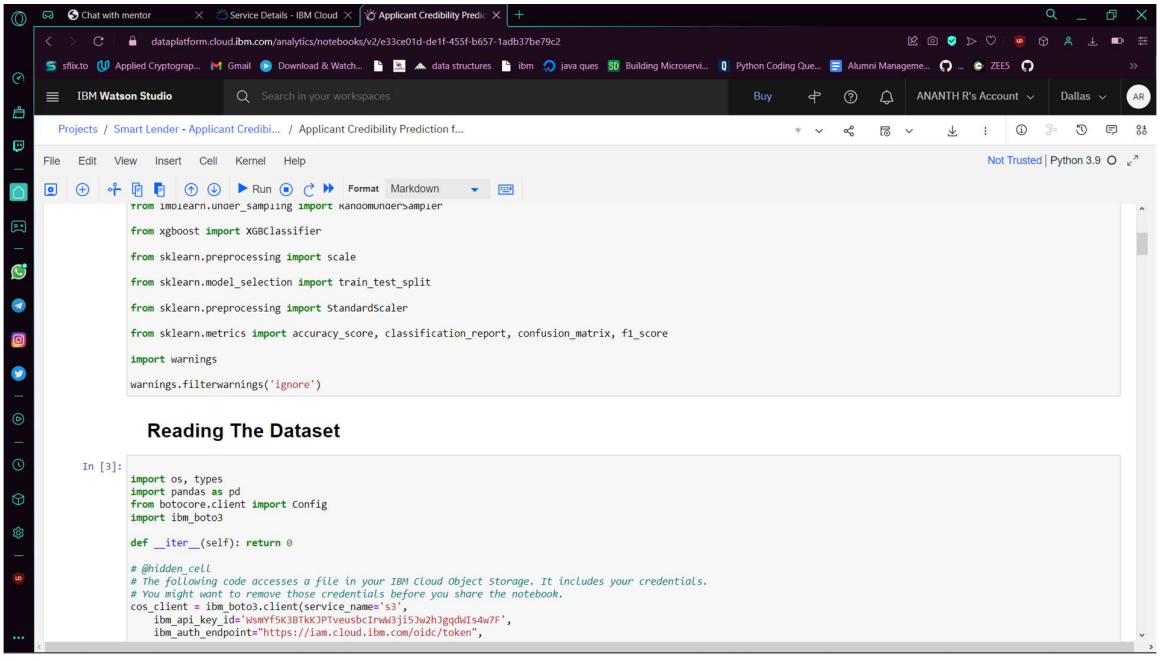
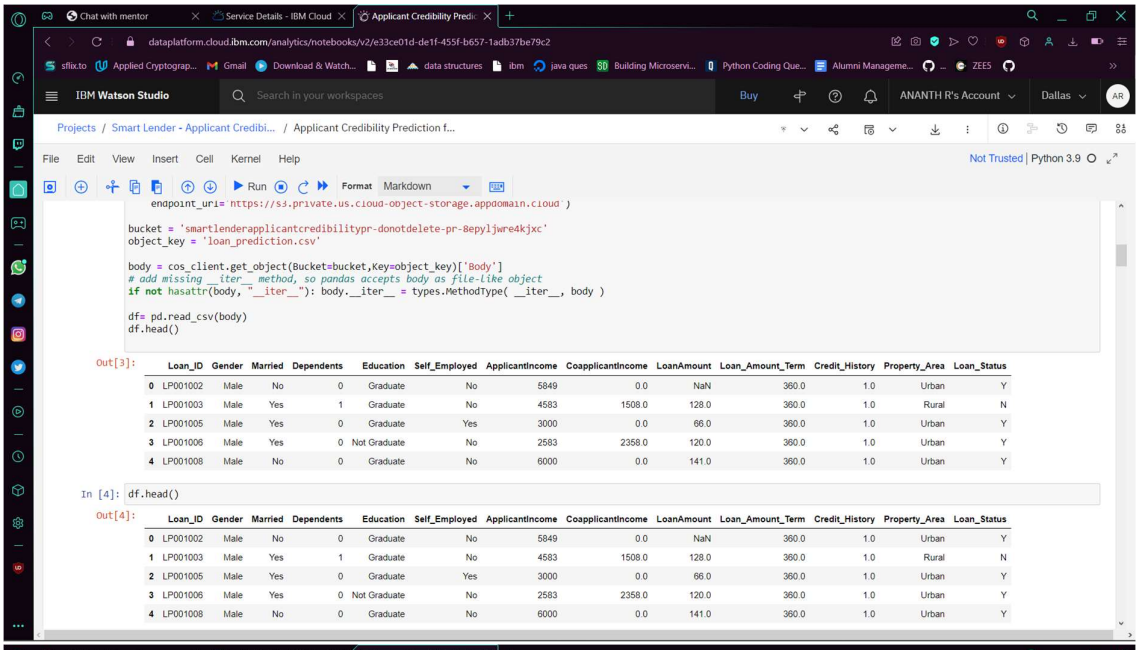
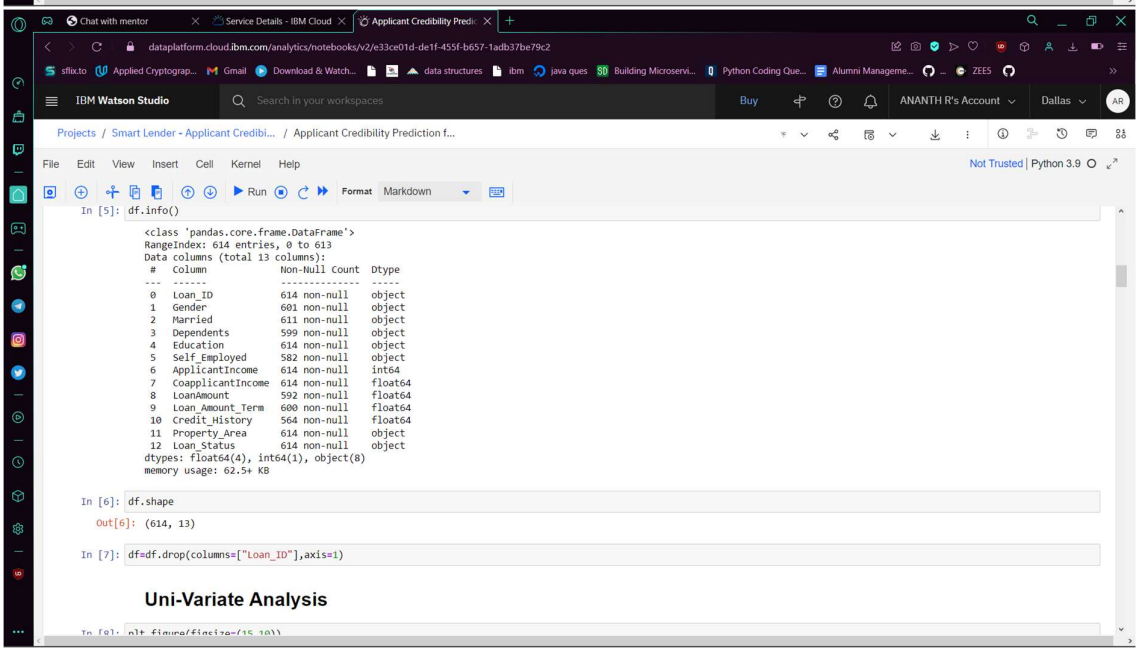
TRAIN THE MODEL ON IBM






Chat with mentor
Service Details - IBM Cloud
Applicant Credibility Predic
dataplatform.cloud.ibm.com/analytics/notebooks/v2/e33ce01d-de1f-455f-b657-1adb37be79c2
IBM Watson Studio
Search in your workspaces
Buy
ANANTH R's Account
Dallas
Projects / Smart Lender - Applicant Credibi... / Applicant Credibility Prediction f...
File Edit View Insert Cell Kernel Help
Not Trusted | Python 3.9
Run
Format Markdown
endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')
bucket = 'smartlenderapplicantcredibilitypr-donotdelete-pr-8epyljwre4kjxc'
object_key = 'loan_prediction.csv'
body = cos_client.get_object(Bucket=bucket,Key=object_key)['Body']
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType(__iter__, body)
df= pd.read_csv(body)
df.head()
Out[3]:
Loan_ID Gender Married Dependents Education Self_Employed ApplicantIncome CoapplicantIncome LoanAmount Loan_Amount_Term Credit_History Property_Area Loan_Status
0 LP001002 Male No 0 Graduate No 5849 0.0 NaN 360.0 1.0 Urban Y
1 LP001003 Male Yes 1 Graduate No 4583 1508.0 128.0 360.0 1.0 Rural N
2 LP001005 Male Yes 0 Graduate Yes 3000 0.0 66.0 360.0 1.0 Urban Y
3 LP001006 Male Yes 0 Not Graduate No 2583 2358.0 120.0 360.0 1.0 Urban Y
4 LP001008 Male No 0 Graduate No 6000 0.0 141.0 360.0 1.0 Urban Y
In [4]: df.head()
Out[4]:
Loan_ID Gender Married Dependents Education Self_Employed ApplicantIncome CoapplicantIncome LoanAmount Loan_Amount_Term Credit_History Property_Area Loan_Status
0 LP001002 Male No 0 Graduate No 5849 0.0 NaN 360.0 1.0 Urban Y
1 LP001003 Male Yes 1 Graduate No 4583 1508.0 128.0 360.0 1.0 Rural N
2 LP001005 Male Yes 0 Graduate Yes 3000 0.0 66.0 360.0 1.0 Urban Y
3 LP001006 Male Yes 0 Not Graduate No 2583 2358.0 120.0 360.0 1.0 Urban Y
4 LP001008 Male No 0 Graduate No 6000 0.0 141.0 360.0 1.0 Urban Y


Chat with mentor
Service Details - IBM Cloud
Applicant Credibility Predic
dataplatform.cloud.ibm.com/analytics/notebooks/v2/e33ce01d-de1f-455f-b657-1adb37be79c2
IBM Watson Studio
Search in your workspaces
Buy
ANANTH R's Account
Dallas
Projects / Smart Lender - Applicant Credibi... / Applicant Credibility Prediction f...
File Edit View Insert Cell Kernel Help
Not Trusted | Python 3.9
Run
Format Markdown
In [5]: df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 614 entries, 0 to 613
Data columns (total 13 columns):
Column Non-Null Count Dtype
--- ------ -------------- -----
0 Loan_ID 614 non-null object
1 Gender 601 non-null object
2 Married 611 non-null object
3 Dependents 599 non-null object
4 Education 614 non-null object
5 Self_Employed 582 non-null object
6 ApplicantIncome 614 non-null int64
7 CoapplicantIncome 614 non-null float64
8 LoanAmount 592 non-null float64
9 Loan_Amount_Term 600 non-null float64
10 Credit_History 564 non-null float64
11 Property_Area 614 non-null object
12 Loan_Status 614 non-null object
dtypes: float64(4), int64(1), object(8)
memory usage: 62.5+ KB
In [6]: df.shape
Out[6]: (614, 13)
In [7]: df=df.drop(columns=["Loan_ID"],axis=1)
Uni-Variate Analysis
In [8]: plt.figure(figsize=(15,10))

Chat with mentor | Service Details - IBM Cloud | Applicant Credibility Predic

dataplatform.cloud.ibm.com/analytics/notebooks/v2/e33ce01d-de1f-455f-b657-1adb37be79c2

IBM Watson Studio | Search in your workspaces | Buy | ANANTH R's Account | Dallas

Projects / Smart Lender - Applicant Credibi... / Applicant Credibility Prediction f...

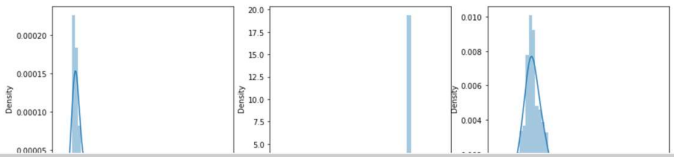
File Edit View Insert Cell Kernel Help | Not Trusted | Python 3.9

Run | Format Markdown

Uni-Variate Analysis

In [8]:
```
plt.figure(figsize=(15,10))
plt.subplot(2,3,1)
sns.distplot(df["ApplicantIncome"])
plt.subplot(2,3,2)
sns.distplot(df["Credit_History"])
plt.subplot(2,3,3)
sns.distplot(df["LoanAmount"])
plt.subplot(2,3,4)
sns.distplot(df["Loan_Amount_Term"])
plt.subplot(2,3,5)
sns.distplot(df["CoapplicantIncome"])
plt.suptitle("NUMERICAL VALUE-DIST PLOT")
plt.show()
```

NUMERICAL VALUE-DIST PLOT



Chat with mentor | Service Details - IBM Cloud | Applicant Credibility Predic

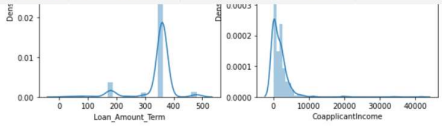
dataplatform.cloud.ibm.com/analytics/notebooks/v2/e33ce01d-de1f-455f-b657-1adb37be79c2

IBM Watson Studio | Search in your workspaces | Buy | ANANTH R's Account | Dallas

Projects / Smart Lender - Applicant Credibi... / Applicant Credibility Prediction f...

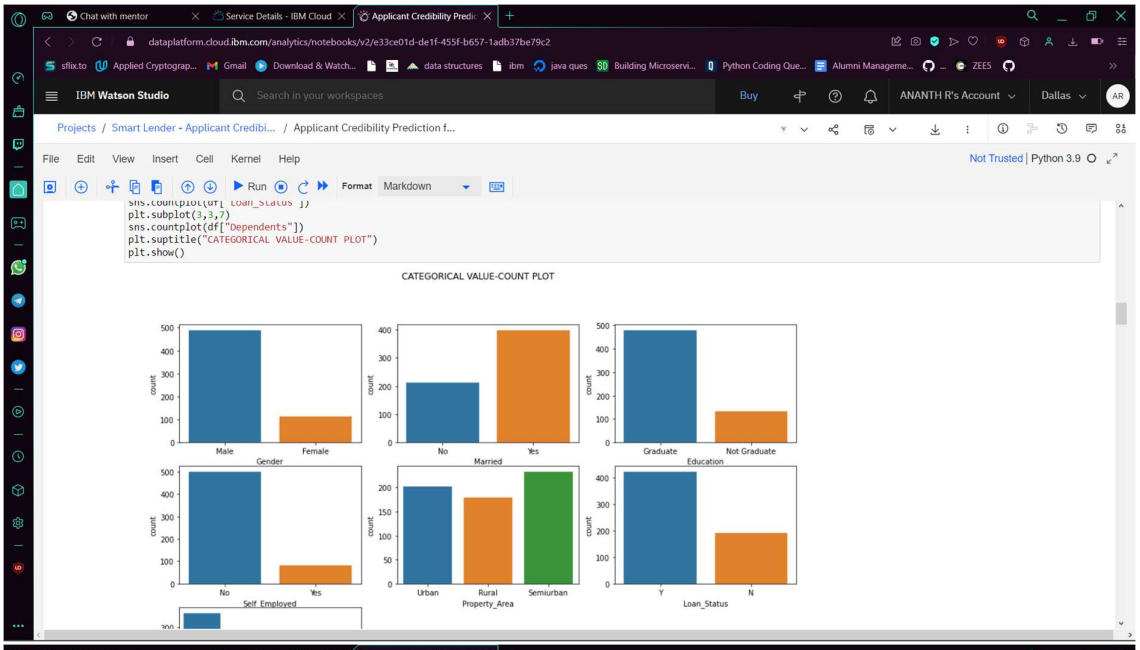
File Edit View Insert Cell Kernel Help | Not Trusted | Python 3.9

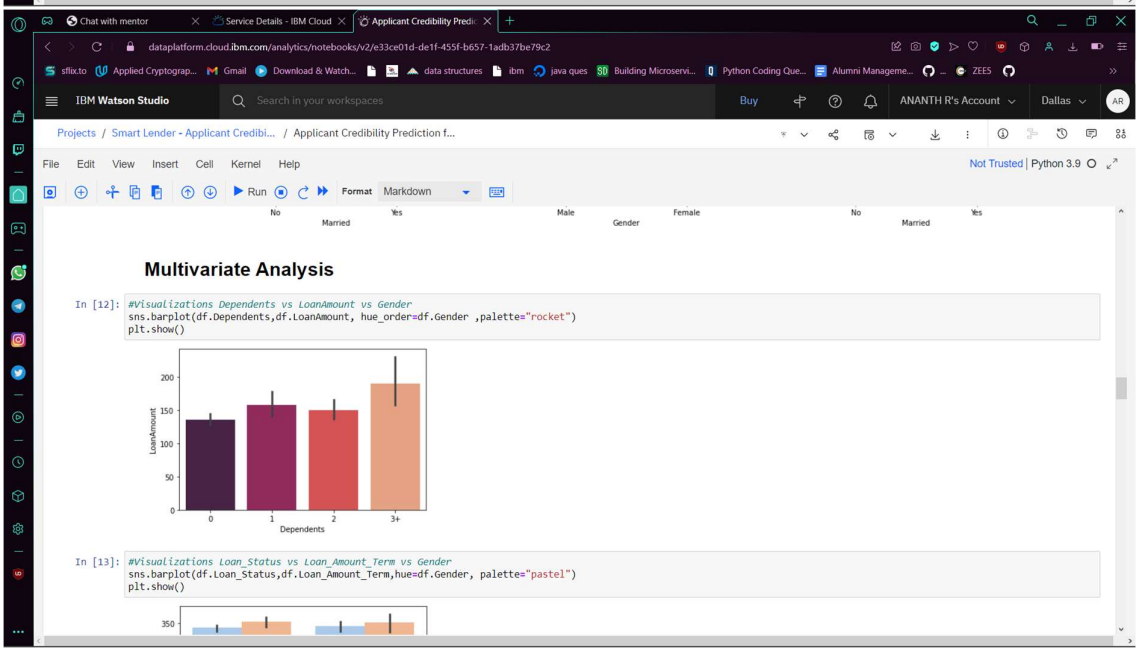
Run | Format Markdown

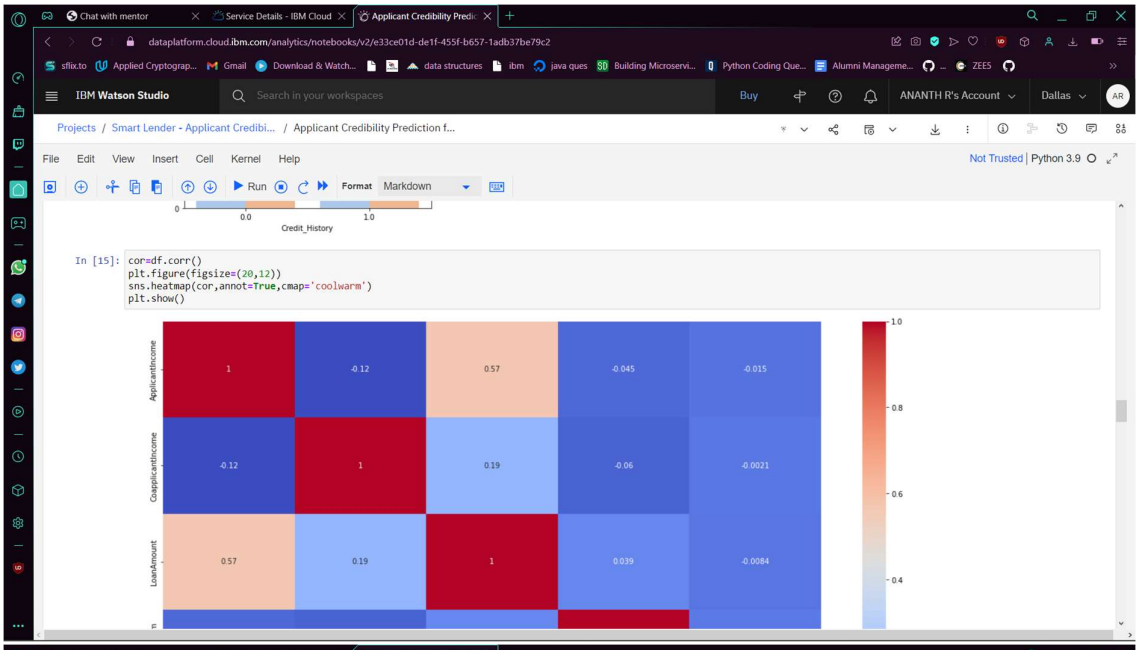


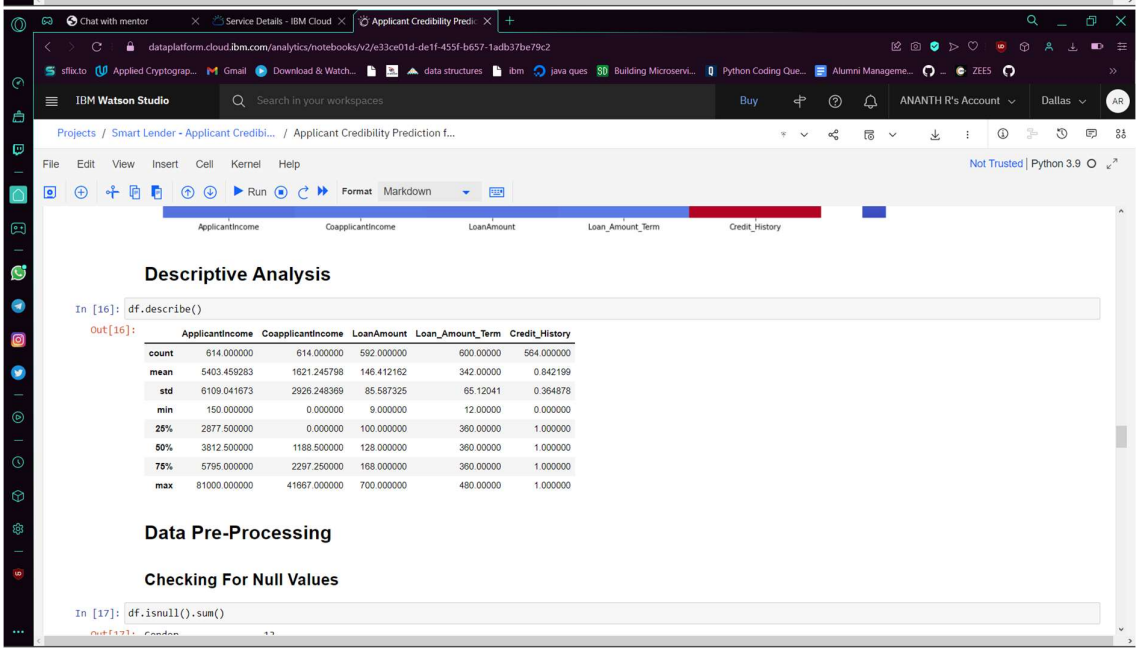
In [9]:
```
plt.figure(figsize=(15,10))
plt.subplot(3,3,1)
sns.countplot(df["Gender"])
plt.subplot(3,3,2)
sns.countplot(df["Married"])
plt.subplot(3,3,3)
sns.countplot(df["Education"])
plt.subplot(3,3,4)
sns.countplot(df["Self_Employed"])
plt.subplot(3,3,5)
sns.countplot(df["Property_Area"])
plt.subplot(3,3,6)
sns.countplot(df["Loan_Status"])
plt.subplot(3,3,7)
sns.countplot(df["Dependents"])
plt.suptitle("CATEGORICAL VALUE-COUNT PLOT")
plt.show()
```

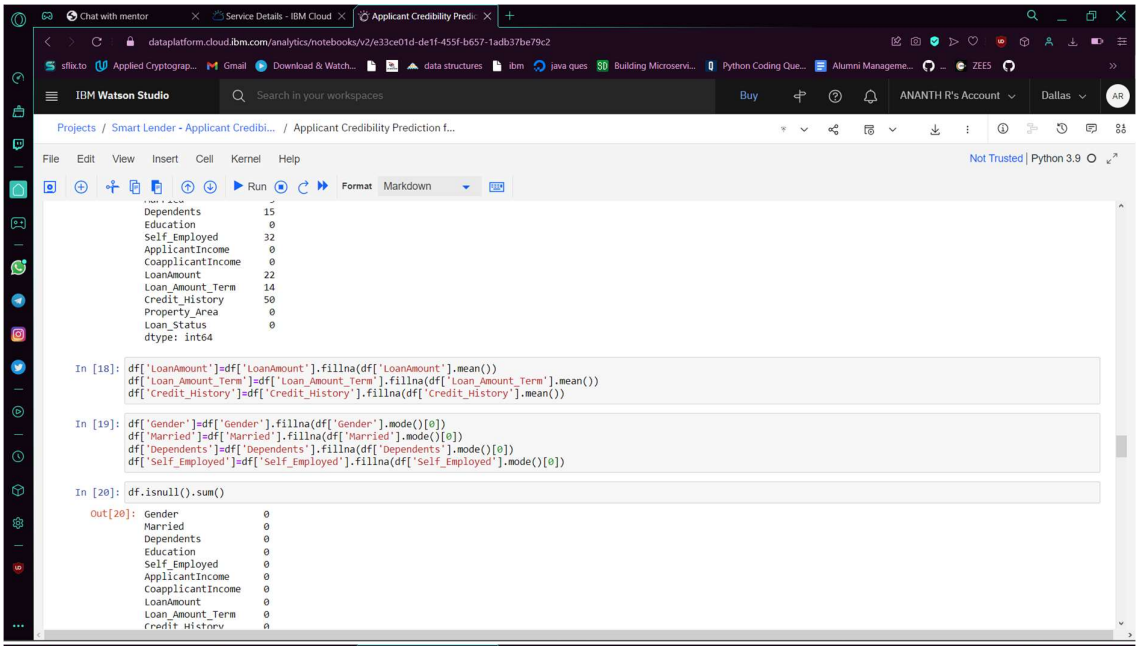
CATEGORICAL VALUE-COUNT PLOT


Chat with mentor
Service Details - IBM Cloud
Applicant Credibility Predi
dataplatform.cloud.ibm.com/analytics/notebooks/v2/e33ce01d-de1f-455f-b657-1adb37be79c2
IBM Watson Studio
Search in your workspaces
Buy
ANANTH R's Account
Dallas
Projects / Smart Lender - Applicant Credibi... / Applicant Credibility Prediction f...
File Edit View Insert Cell Kernel Help
Not Trusted | Python 3.9
Run
Format Markdown
plt.subplot(3,3,7)
sns.countplot(df["Dependents"])
plt.suptitle("CATEGORICAL VALUE-COUNT PLOT")
plt.show()
CATEGORICAL VALUE-COUNT PLOT
count
Male
Female
Gender
No
Yes
Married
Graduate
Not Graduate
Education
No
Yes
Self_Employed
Urban
Rural
Semiurban
Property_Area
Y
N
Loan_Status


Chat with mentor
Service Details - IBM Cloud
Applicant Credibility Predi
dataplatform.cloud.ibm.com/analytics/notebooks/v2/e33ce01d-de1f-455f-b657-1adb37be79c2
IBM Watson Studio
Search in your workspaces
Buy
ANANTH R's Account
Dallas
Projects / Smart Lender - Applicant Credibi... / Applicant Credibility Prediction f...
File Edit View Insert Cell Kernel Help
Not Trusted | Python 3.9
Run
Format Markdown
No
Yes
Married
Male
Female
Gender
No
Yes
Married
Multivariate Analysis
In [12]:
#Visualizations Dependents vs LoanAmount vs Gender
sns.barplot(df.Dependents,df.LoanAmount, hue_order=df.Gender ,palette="rocket")
plt.show()
LoanAmount
0
1
2
3+
Dependents
In [13]:
#Visualizations Loan_Status vs Loan_Amount_Term vs Gender
sns.barplot(df.Loan_Status,df.Loan_Amount_Term,hue=df.Gender, palette="pastel")
plt.show()


Chat with mentor
Service Details - IBM Cloud
Applicant Credibility Predi
dataplatform.cloud.ibm.com/analytics/notebooks/v2/e33ce01d-de1f-455f-b657-1adb37be79c2
IBM Watson Studio
Search in your workspaces
Buy
ANANTH R's Account
Dallas
Projects / Smart Lender - Applicant Credibi... / Applicant Credibility Prediction f...
File Edit View Insert Cell Kernel Help
Not Trusted | Python 3.9
Run
Format Markdown
Credit_History
In [15]:
cor=df.corr()
plt.figure(figsize=(20,12))
sns.heatmap(cor,annot=True,cmap='coolwarm')
plt.show()
ApplicantIncome
CoapplicantIncome
LoanAmount
1
-0.12
0.57
-0.045
-0.015
-0.12
1
0.19
-0.06
-0.0021
0.57
0.19
1
0.039
-0.0084


Chat with mentor
Service Details - IBM Cloud
Applicant Credibility Predi
dataplatform.cloud.ibm.com/analytics/notebooks/v2/e33ce01d-de1f-455f-b657-1adb37be79c2
IBM Watson Studio
Search in your workspaces
Buy
ANANTH R's Account
Dallas
Projects / Smart Lender - Applicant Credibi... / Applicant Credibility Prediction f...
File Edit View Insert Cell Kernel Help
Not Trusted | Python 3.9
Run
Format Markdown
ApplicantIncome
CoapplicantIncome
LoanAmount
Loan_Amount_Term
Credit_History
Descriptive Analysis
In [16]:
df.describe()
Out[16]:
ApplicantIncome CoapplicantIncome LoanAmount Loan_Amount_Term Credit_History
count 614.000000 614.000000 592.000000 600.00000 564.000000
mean 5403.459283 1621.245798 146.412162 342.00000 0.842199
std 6109.041673 2926.248369 85.587325 65.12041 0.364878
min 150.000000 0.000000 9.000000 12.00000 0.000000
25% 2877.500000 0.000000 100.000000 360.00000 1.000000
50% 3812.500000 1188.500000 128.000000 360.00000 1.000000
75% 5795.000000 2297.250000 168.000000 360.00000 1.000000
max 81000.000000 41667.000000 700.000000 480.00000 1.000000
Data Pre-Processing
Checking For Null Values
In [17]:
df.isnull().sum()



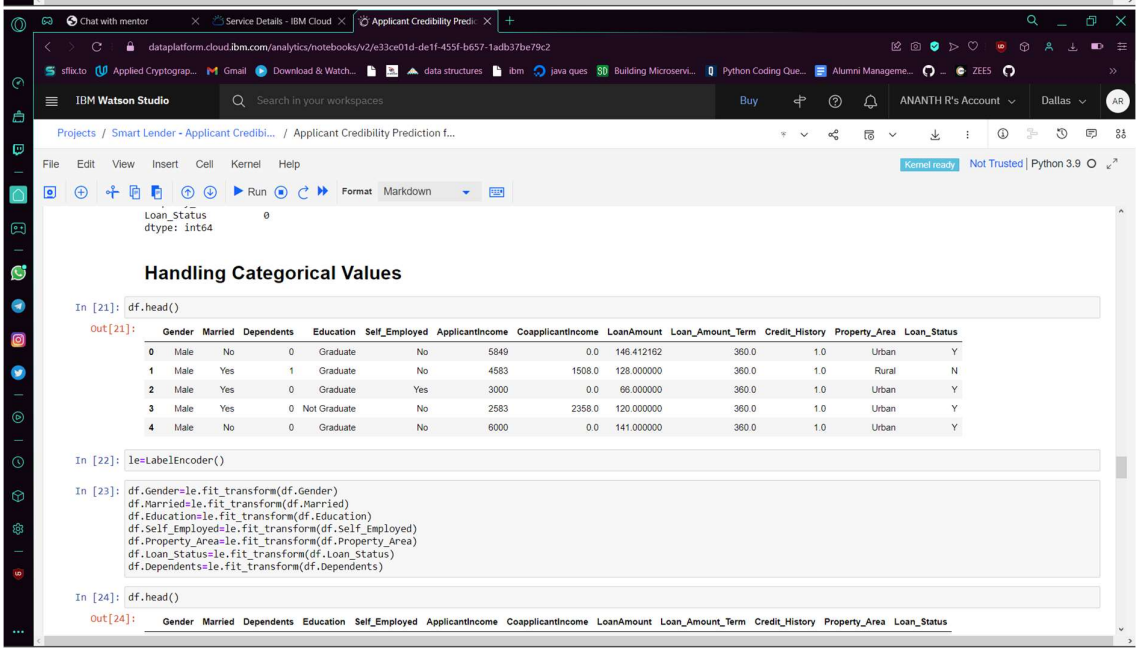
```
Dependents          15
Education            0
Self_Employed       32
ApplicantIncome      0
CoapplicantIncome    0
LoanAmount          22
Loan_Amount_Term    14
Credit_History      50
Property_Area        0
Loan_Status          0
dtype: int64
```

In [18]:
```
df['LoanAmount']=df['LoanAmount'].fillna(df['LoanAmount'].mean())
df['Loan_Amount_Term']=df['Loan_Amount_Term'].fillna(df['Loan_Amount_Term'].mean())
df['Credit_History']=df['Credit_History'].fillna(df['Credit_History'].mean())
```

In [19]:
```
df['Gender']=df['Gender'].fillna(df['Gender'].mode()[0])
df['Married']=df['Married'].fillna(df['Married'].mode()[0])
df['Dependents']=df['Dependents'].fillna(df['Dependents'].mode()[0])
df['Self_Employed']=df['Self_Employed'].fillna(df['Self_Employed'].mode()[0])
```

In [20]:
```
df.isnull().sum()
```

Out[20]:
```
Gender              0
Married             0
Dependents          0
Education           0
Self_Employed       0
ApplicantIncome     0
CoapplicantIncome   0
LoanAmount          0
Loan_Amount_Term    0
Credit_History      0
```



```
Loan_Status          0
dtype: int64
```

Handling Categorical Values

In [21]:
```
df.head()
```

Out[21]:

	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status
0	Male	No	0	Graduate	No	5849	0.0	146.412162	360.0	1.0	Urban	Y
1	Male	Yes	1	Graduate	No	4583	1508.0	128.000000	360.0	1.0	Rural	N
2	Male	Yes	0	Graduate	Yes	3000	0.0	66.000000	360.0	1.0	Urban	Y
3	Male	Yes	0	Not Graduate	No	2583	2358.0	120.000000	360.0	1.0	Urban	Y
4	Male	No	0	Graduate	No	6000	0.0	141.000000	360.0	1.0	Urban	Y

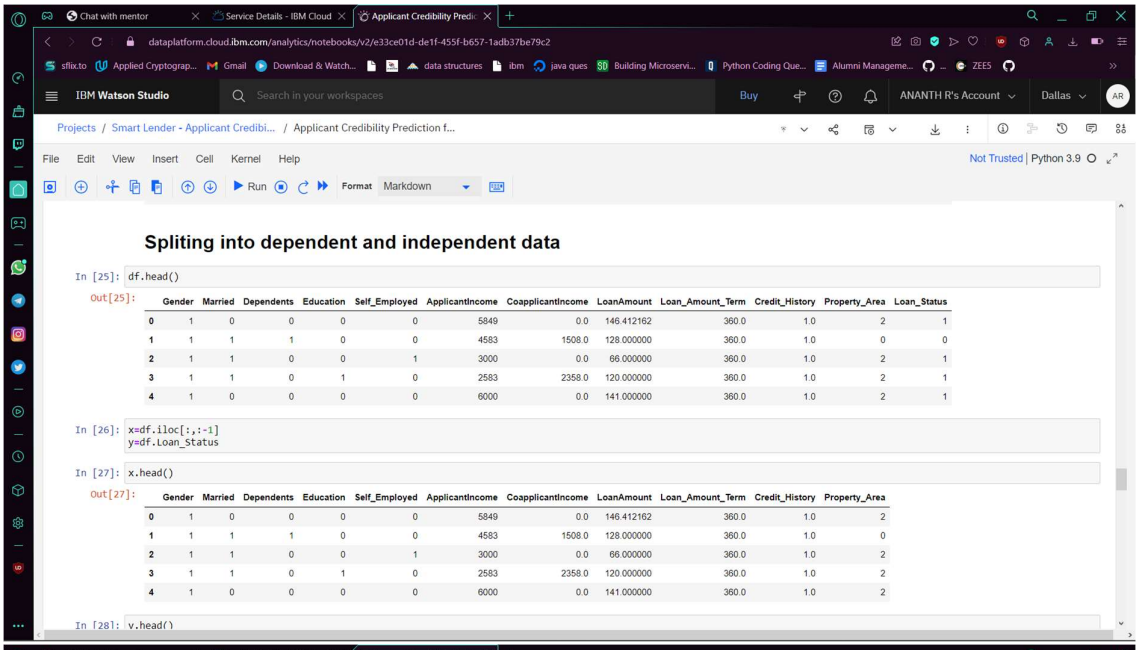
In [22]:
```
le=LabelEncoder()
```

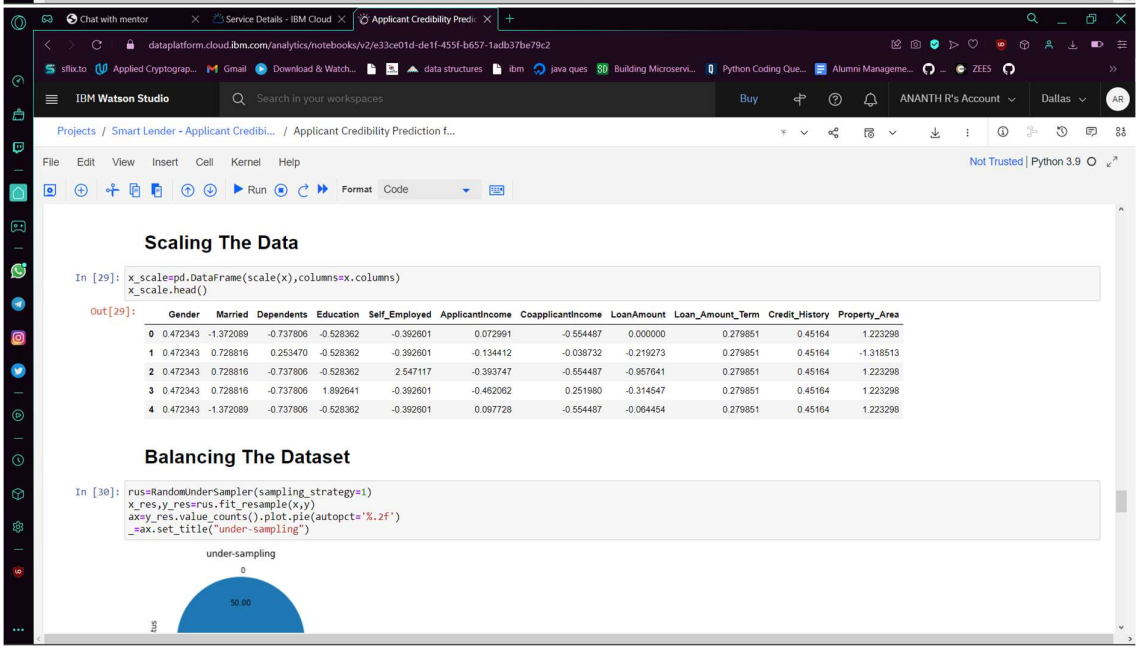
In [23]:
```
df.Gender=le.fit_transform(df.Gender)
df.Married=le.fit_transform(df.Married)
df.Education=le.fit_transform(df.Education)
df.Self_Employed=le.fit_transform(df.Self_Employed)
df.Property_Area=le.fit_transform(df.Property_Area)
df.Loan_Status=le.fit_transform(df.Loan_Status)
df.Dependents=le.fit_transform(df.Dependents)
```

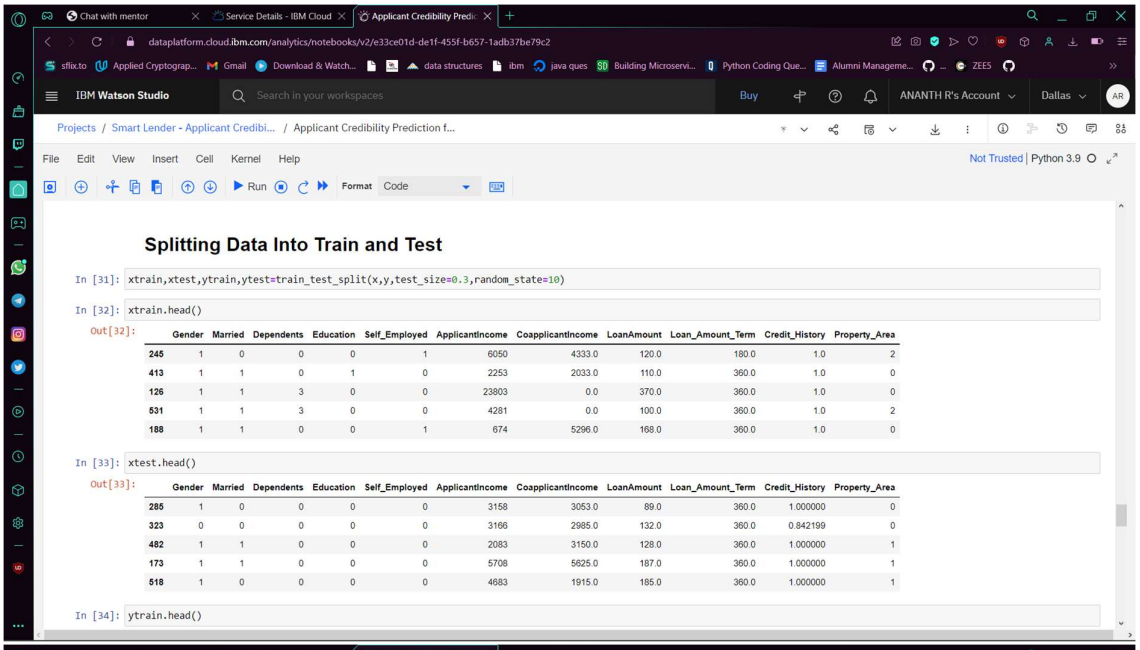
In [24]:
```
df.head()
```

Out[24]:

Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status

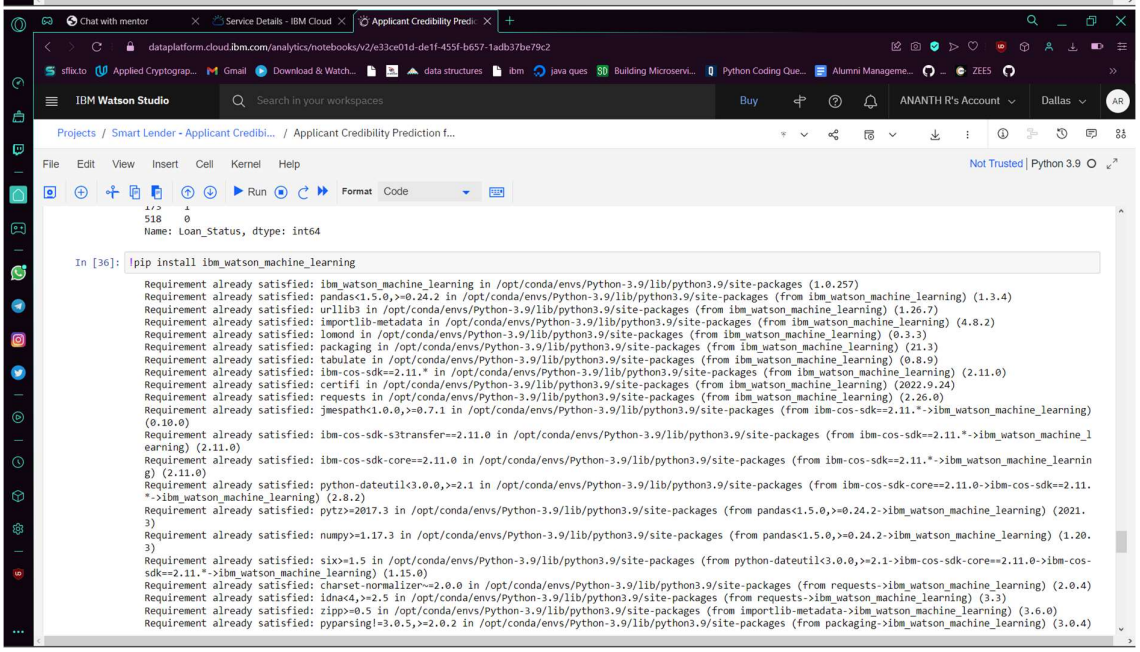

Chat with mentor
Service Details - IBM Cloud
Applicant Credibility Predic
dataplatform.cloud.ibm.com/analytics/notebooks/v2/e33ce01d-de1f-455f-b657-1adb37be79c2
IBM Watson Studio
Search in your workspaces
Buy
ANANTH R's Account
Dallas
Projects / Smart Lender - Applicant Credibi... / Applicant Credibility Prediction f...
File Edit View Insert Cell Kernel Help
Not Trusted | Python 3.9
Run
Format Markdown
Spliting into dependent and independent data
In [25]: df.head()
Out[25]:
Gender Married Dependents Education Self_Employed ApplicantIncome CoapplicantIncome LoanAmount Loan_Amount_Term Credit_History Property_Area Loan_Status
0 1 0 0 0 0 5849 0.0 146.412162 360.0 1.0 2 1
1 1 1 1 0 0 4583 1508.0 128.000000 360.0 1.0 0 0
2 1 1 0 0 1 3000 0.0 66.000000 360.0 1.0 2 1
3 1 1 0 1 0 2583 2358.0 120.000000 360.0 1.0 2 1
4 1 0 0 0 0 6000 0.0 141.000000 360.0 1.0 2 1
In [26]: x=df.iloc[:,:-1]
y=df.Loan_Status
In [27]: x.head()
Out[27]:
Gender Married Dependents Education Self_Employed ApplicantIncome CoapplicantIncome LoanAmount Loan_Amount_Term Credit_History Property_Area
0 1 0 0 0 0 5849 0.0 146.412162 360.0 1.0 2
1 1 1 1 0 0 4583 1508.0 128.000000 360.0 1.0 0
2 1 1 0 0 1 3000 0.0 66.000000 360.0 1.0 2
3 1 1 0 1 0 2583 2358.0 120.000000 360.0 1.0 2
4 1 0 0 0 0 6000 0.0 141.000000 360.0 1.0 2
In [28]: y.head()

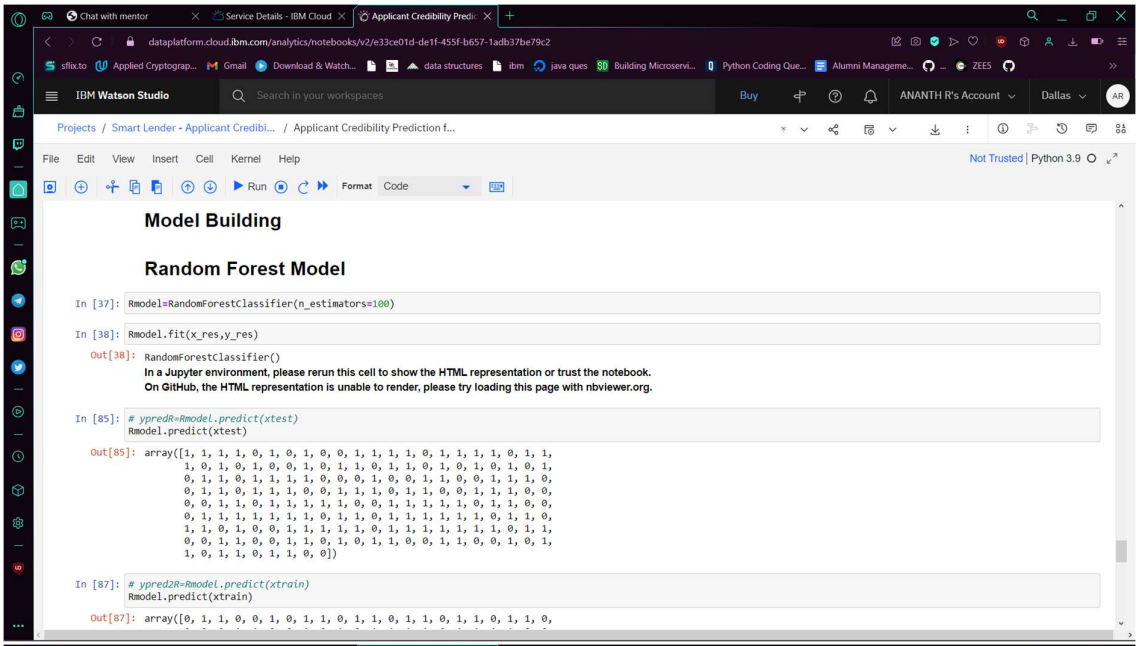

Chat with mentor
Service Details - IBM Cloud
Applicant Credibility Predic
dataplatform.cloud.ibm.com/analytics/notebooks/v2/e33ce01d-de1f-455f-b657-1adb37be79c2
IBM Watson Studio
Search in your workspaces
Buy
ANANTH R's Account
Dallas
Projects / Smart Lender - Applicant Credibi... / Applicant Credibility Prediction f...
File Edit View Insert Cell Kernel Help
Not Trusted | Python 3.9
Run
Format Code
Scaling The Data
In [29]: x_scale=pd.DataFrame(scale(x),columns=x.columns)
x_scale.head()
Out[29]:
Gender Married Dependents Education Self_Employed ApplicantIncome CoapplicantIncome LoanAmount Loan_Amount_Term Credit_History Property_Area
0 0.472343 -1.372089 -0.737806 -0.528362 -0.392601 0.072991 -0.554487 0.000000 0.279851 0.45164 1.223298
1 0.472343 0.728816 0.253470 -0.528362 -0.392601 -0.134412 -0.038732 -0.219273 0.279851 0.45164 -1.318513
2 0.472343 0.728816 -0.737806 -0.528362 2.547117 -0.393747 -0.554487 -0.957641 0.279851 0.45164 1.223298
3 0.472343 0.728816 -0.737806 1.892641 -0.392601 -0.462062 0.251980 -0.314547 0.279851 0.45164 1.223298
4 0.472343 -1.372089 -0.737806 -0.528362 -0.392601 0.097728 -0.554487 -0.064454 0.279851 0.45164 1.223298
Balancing The Dataset
In [30]: rus=RandomUnderSampler(sampling_strategy=1)
x_res,y_res=rus.fit_resample(x,y)
ax=y_res.value_counts().plot.pie(autopct='%.2f')
_=ax.set_title("under-sampling")
under-sampling
0
50.00

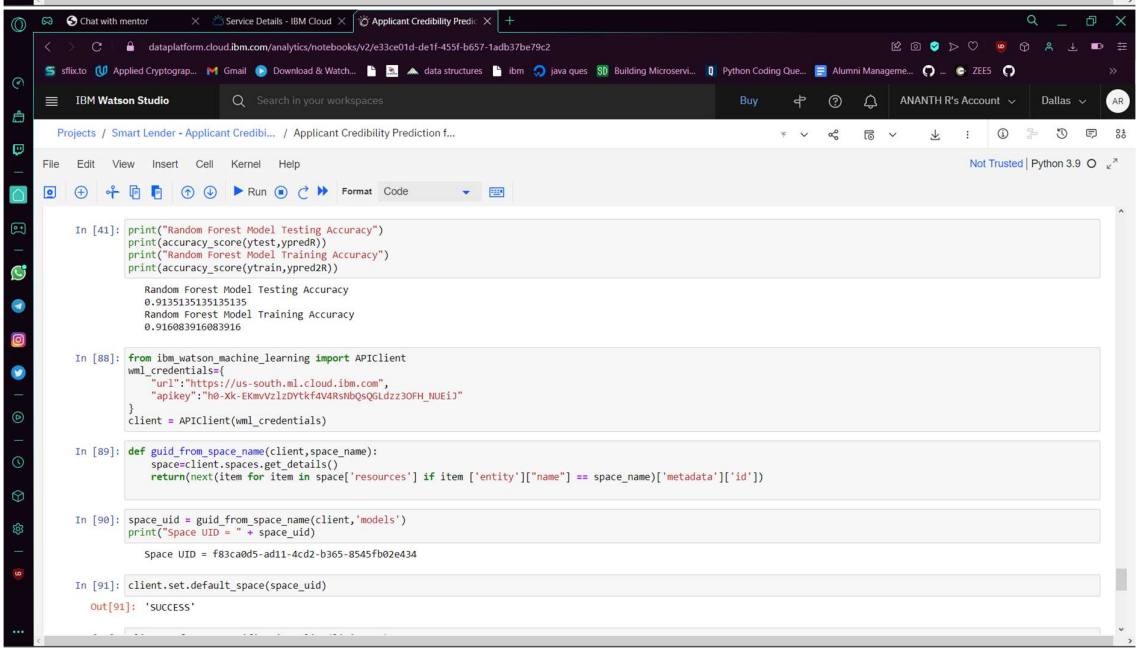


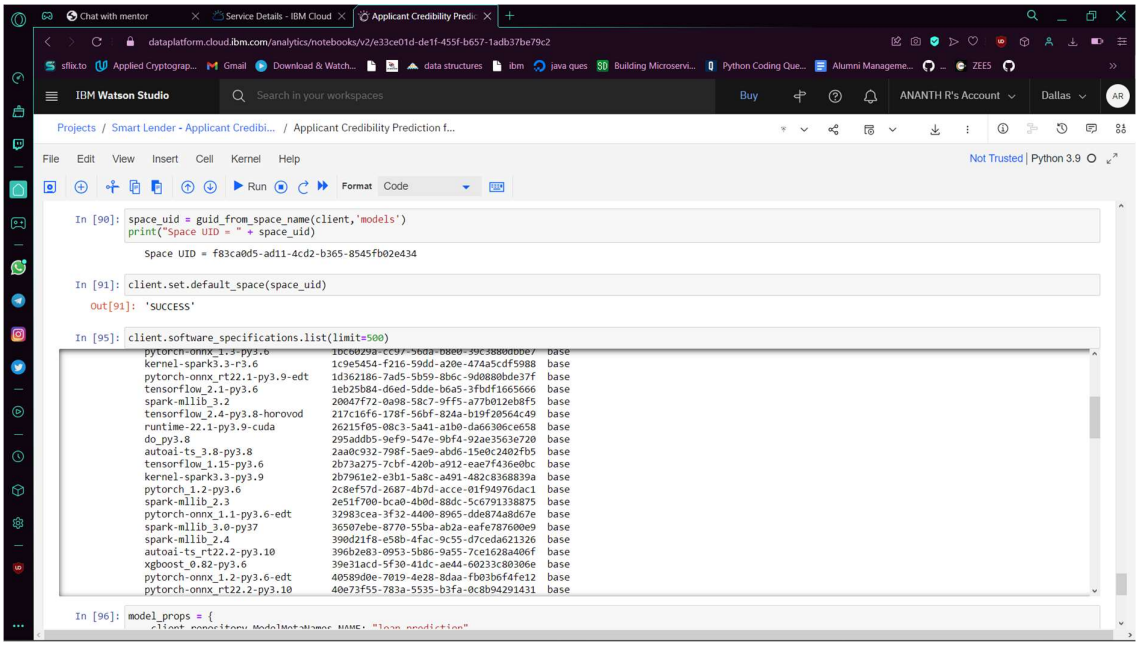
	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area
245	1	0	0	0	1	6050	4333.0	120.0	180.0	1.0	2
413	1	1	0	1	0	2253	2033.0	110.0	360.0	1.0	0
126	1	1	3	0	0	23803	0.0	370.0	360.0	1.0	0
531	1	1	3	0	0	4281	0.0	100.0	360.0	1.0	2
188	1	1	0	0	1	674	5296.0	168.0	360.0	1.0	0

	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area
285	1	0	0	0	0	3158	3053.0	89.0	360.0	1.000000	0
323	0	0	0	0	0	3166	2985.0	132.0	360.0	0.842199	0
482	1	1	0	0	0	2083	3150.0	128.0	360.0	1.000000	1
173	1	1	0	0	0	5708	5625.0	187.0	360.0	1.000000	1
518	1	0	0	0	0	4683	1915.0	185.0	360.0	1.000000	1




Chat with mentor
Service Details - IBM Cloud
Applicant Credibility Predi
dataplatform.cloud.ibm.com/analytics/notebooks/v2/e33ce01d-de1f-455f-b657-1adb37be79c2
IBM Watson Studio
Search in your workspaces
Buy
ANANTH R's Account
Dallas
Projects / Smart Lender - Applicant Credibi... / Applicant Credibility Prediction f...
File Edit View Insert Cell Kernel Help
Not Trusted | Python 3.9
Run
Format Code
Model Building
Random Forest Model
In [37]: Rmodel=RandomForestClassifier(n_estimators=100)
In [38]: Rmodel.fit(x_res,y_res)
Out[38]: RandomForestClassifier()
In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.
In [85]: # ypredR=Rmodel.predict(xtest)
Rmodel.predict(xtest)
Out[85]: array([1, 1, 1, 1, 0, 1, 0, 1, 0, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1,
1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1,
0, 1, 1, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 1, 0,
0, 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 0, 1, 1, 0, 0, 1, 1, 1, 0, 0,
0, 0, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 0, 1, 1, 0, 0,
0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0,
1, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1,
0, 0, 1, 1, 0, 0, 1, 1, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 0, 1,
1, 0, 1, 1, 0, 1, 1, 0, 0])
In [87]: # ypred2R=Rmodel.predict(xtrain)
Rmodel.predict(xtrain)
Out[87]: array([0, 1, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0,


Chat with mentor
Service Details - IBM Cloud
Applicant Credibility Predi
dataplatform.cloud.ibm.com/analytics/notebooks/v2/e33ce01d-de1f-455f-b657-1adb37be79c2
IBM Watson Studio
Search in your workspaces
Buy
ANANTH R's Account
Dallas
Projects / Smart Lender - Applicant Credibi... / Applicant Credibility Prediction f...
File Edit View Insert Cell Kernel Help
Not Trusted | Python 3.9
Run
Format Code
In [41]: print("Random Forest Model Testing Accuracy")
print(accuracy_score(ytest,ypredR))
print("Random Forest Model Training Accuracy")
print(accuracy_score(ytrain,ypred2R))
Random Forest Model Testing Accuracy
0.9135135135135135
Random Forest Model Training Accuracy
0.916083916083916
In [88]: from ibm_watson_machine_learning import APIClient
wml_credentials={
"url":"https://us-south.ml.cloud.ibm.com",
"apikey":"h0-Xk-EKmvVzlzDYtkf4V4RsNbQsQGLdzz3OFH_NUEiJ"
}
client = APIClient(wml_credentials)
In [89]: def guid_from_space_name(client,space_name):
space=client.spaces.get_details()
return(next(item for item in space['resources'] if item ['entity']["name"] == space_name)['metadata']['id'])
In [90]: space_uid = guid_from_space_name(client,'models')
print("Space UID = " + space_uid)
Space UID = f83ca0d5-ad11-4cd2-b365-8545fb02e434
In [91]: client.set.default_space(space_uid)
Out[91]: 'SUCCESS'


Chat with mentor
Service Details - IBM Cloud
Applicant Credibility Predi
dataplatform.cloud.ibm.com/analytics/notebooks/v2/e3ce01d-de1f-455f-b657-1adb37be79c2
IBM Watson Studio
Search in your workspaces
Buy
ANANTH R's Account
Dallas
Projects / Smart Lender - Applicant Credibi... / Applicant Credibility Prediction f...
File Edit View Insert Cell Kernel Help
Not Trusted | Python 3.9
Run
Format Code
In [90]: space_uid = guid_from_space_name(client,'models')
print("Space UID = " + space_uid)
Space UID = f83ca0d5-ad11-4cd2-b365-8545fb02e434
In [91]: client.set.default_space(space_uid)
Out[91]: 'SUCCESS'
In [95]: client.software_specifications.list(limit=500)
kernel-spark3.3-r3.6 1c9e5454-f216-59dd-a20e-474a5cdf5988 base
pytorch-onnx_rt22.1-py3.9-edt 1d362186-7ad5-5b59-8b6c-9d0880bde37f base
tensorflow_2.1-py3.6 1eb25b84-d6ed-5dde-b6a5-3fbdf1665666 base
spark-mllib_3.2 20047f72-0a98-58c7-9ff5-a77b012eb8f5 base
tensorflow_2.4-py3.8-horovod 217c16f6-178f-56bf-824a-b19f20564c49 base
runtime-22.1-py3.9-cuda 26215f05-08c3-5a41-a1b0-da66306ce658 base
do_py3.8 295addb5-9ef9-547e-9bf4-92ae3563e720 base
autoai-ts_3.8-py3.8 2aa0c932-798f-5ae9-abd6-15e0c2402fb5 base
tensorflow_1.15-py3.6 2b73a275-7cbf-420b-a912-eae7f436e0bc base
kernel-spark3.3-py3.9 2b7961e2-e3b1-5a8c-a491-482c8368839a base
pytorch_1.2-py3.6 2c8ef57d-2687-4b7d-acce-01f94976dac1 base
spark-mllib_2.3 2e51f700-bca0-4b0d-88dc-5c6791338875 base
pytorch-onnx_1.1-py3.6-edt 32983cea-3f32-4400-8965-dde874a8d67e base
spark-mllib_3.0-py37 36507ebe-8770-55ba-ab2a-eafe787600e9 base
spark-mllib_2.4 390d21f8-e58b-4fac-9c55-d7ceda621326 base
autoai-ts_rt22.2-py3.10 396b2e83-0953-5b86-9a55-7ce1628a406f base
xgboost_0.82-py3.6 39e31acd-5f30-41dc-ae44-60233c80306e base
pytorch-onnx_1.2-py3.6-edt 40589d0e-7019-4e28-8daa-fb03b6f4fe12 base
pytorch-onnx_rt22.2-py3.10 40e73f55-783a-5535-b3fa-0c8b94291431 base
In [96]: model_props = {