Machine laerning

Using Linear Regression to train data set and make predictions

Done by Chukwuemeka @ linkedin.com/in/chukwuemeka-agu

Import Libraries

In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

Import Data set (Note: Data set used is Fictitious)

In [2]:

```
salary = pd.read_csv("Salary_Data.csv")
salary.head()
```

Out[2]:

	YearsExperience	Salary
0	1.1	39343
1	1.3	46205
2	1.5	37731
3	2.0	43525
4	2.2	39891

Difine x(preditor variable), y(target variable)

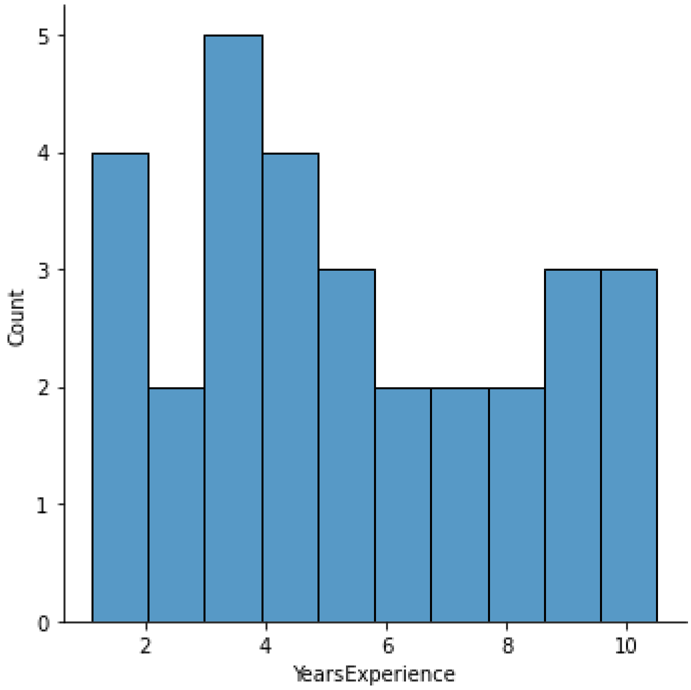
In [3]:

```
X = salary.iloc[:,:-1].values
y = salary.iloc[:,1].values
```

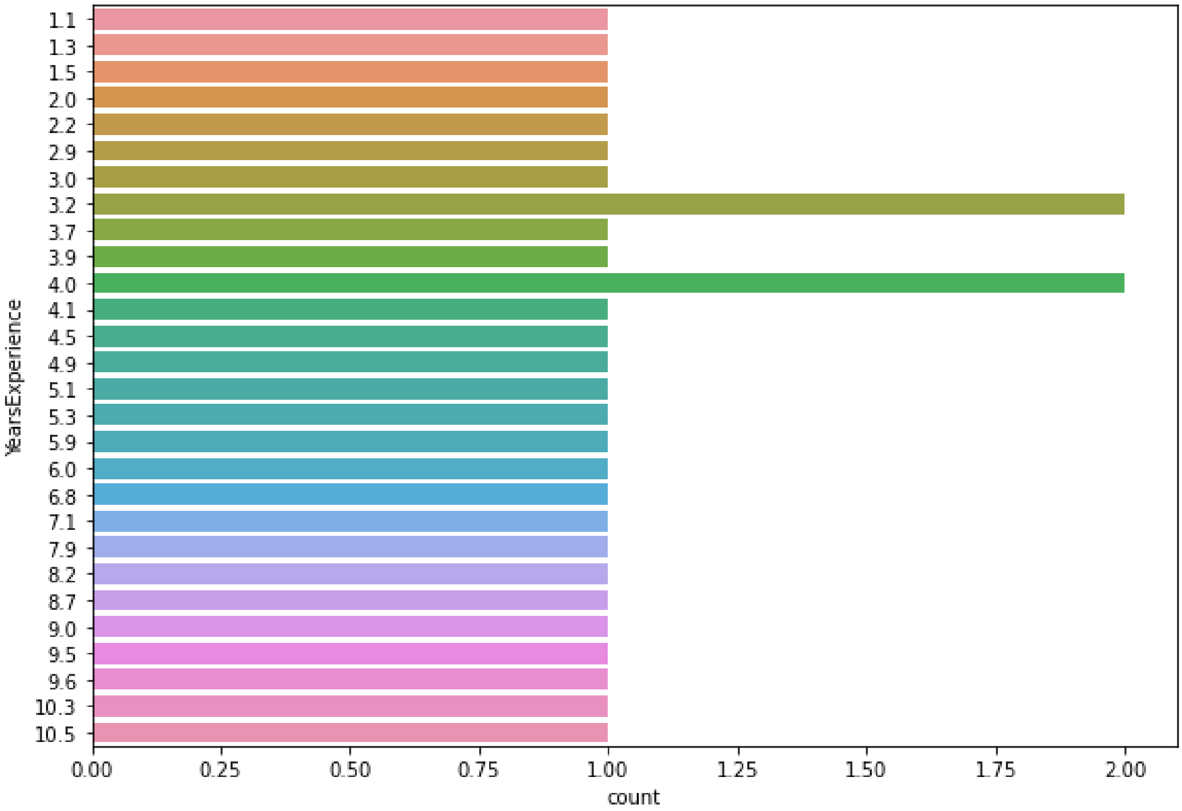
Make some visualizations to understand the data further

In [4]:

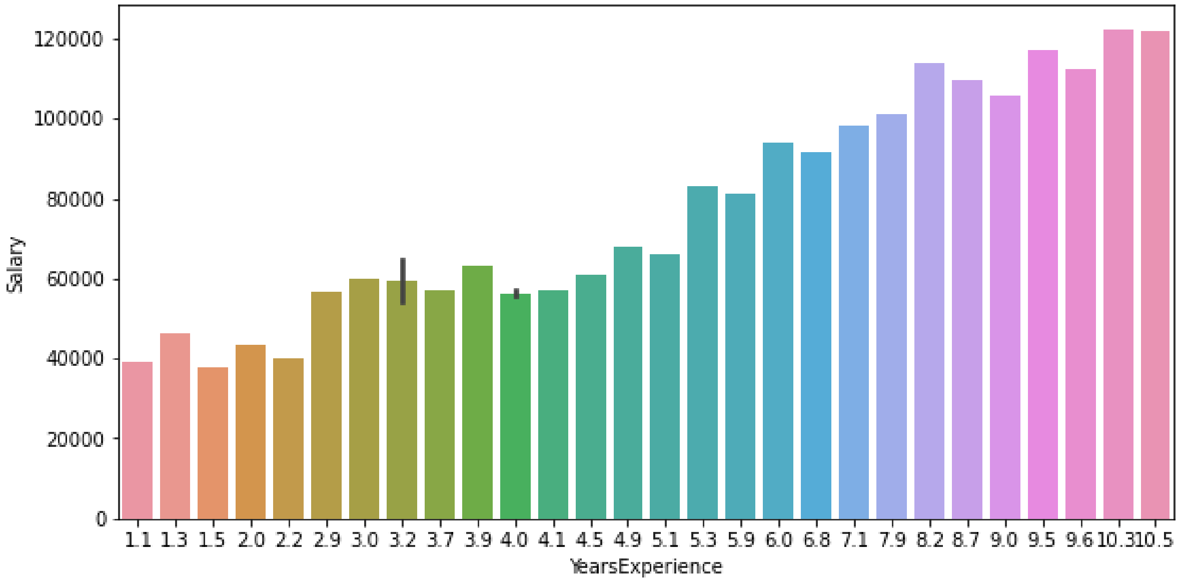
```
sns.displot(salary["YearsExperience"],kde=False, bins=10)
plt.show()
```



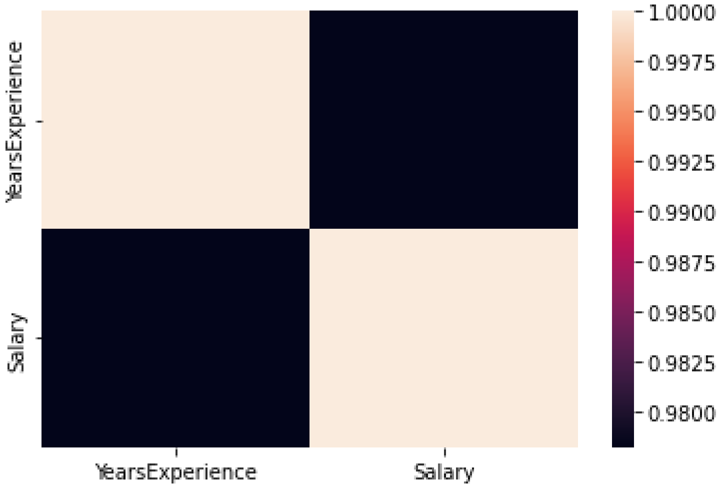
In [5]:
```
plt.figure(figsize=(10,7))
sns.countplot(y="YearsExperience",data = salary)
plt.show()
```



In [6]:
```
plt.figure(figsize=(10,5))
sns.barplot(data=salary,x="YearsExperience",y="Salary")
plt.show()
```



In [7]:
```
sns.heatmap(salary.corr())
plt.show()
```



Import train_test_split

And split the data into training and testing sets

In [8]:
```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=1/3, random_state=0)
```

Import LinearRegression and train the model

In [9]:
```
from sklearn.linear_model import LinearRegression
lr = LinearRegression()
lr.fit(X_train,y_train)
```

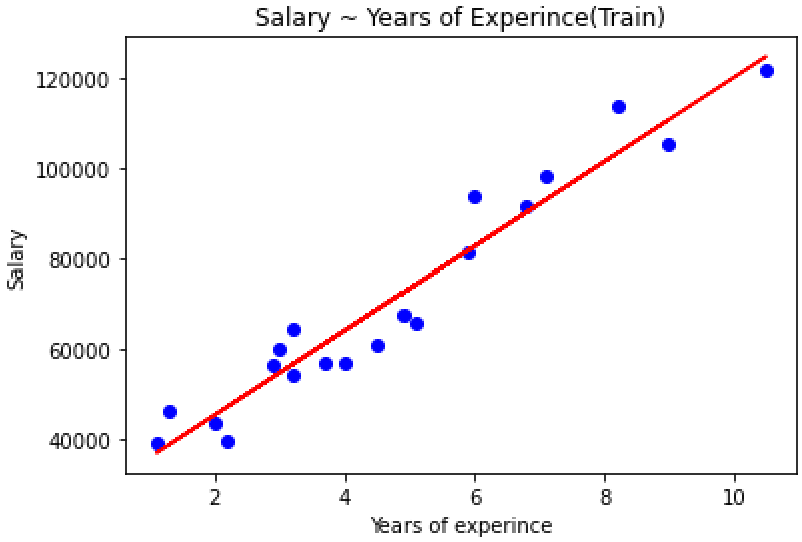
Out[9]:
```
LinearRegression()
```

In [10]:
```
y_pred = lr.predict(X_test)
y_pred
```

Out[10]:
```
array([ 40835.10590871, 123079.39940819,  65134.55626083,  63265.36777221,
       115602.64545369, 108125.8914992 , 116537.23969801,  64199.96201652,
        76349.68719258, 100649.1375447 ])
```

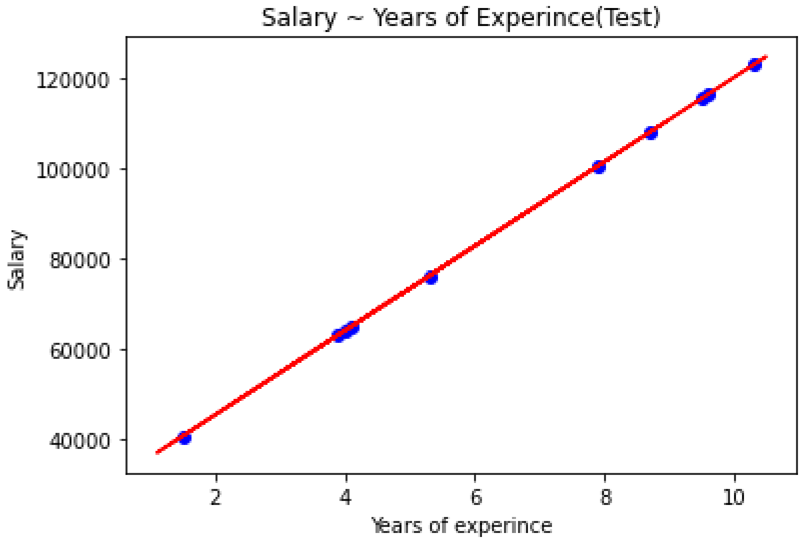
Visualize the train set

In [11]:
```python
plt.scatter(X_train, y_train, color="blue")
plt.plot(X_train, lr.predict(X_train), color="red")
plt.title("Salary ~ Years of Experince(Train)")
plt.ylabel("Salary")
plt.xlabel("Years of experince")
plt.show()
```



Visualize the test set to see how good it fits

In [12]:
```python
plt.scatter(X_test, y_pred, color="blue")
plt.plot(X_train, lr.predict(X_train), color="red")
plt.title("Salary ~ Years of Experince(Test)")
plt.ylabel("Salary")
plt.xlabel("Years of experince")
plt.show()
```



In [13]:
```python
from sklearn import metrics

print("MAE", metrics.mean_absolute_error(y_test,y_pred))
print("MSE", metrics.mean_squared_error(y_test,y_pred))
print("RMSE", np.sqrt(metrics.mean_absolute_error(y_test,y_pred)))
```

```
MAE 3426.4269374307123
MSE 21026037.329511296
RMSE 58.53568943329114
```

In [ ]: