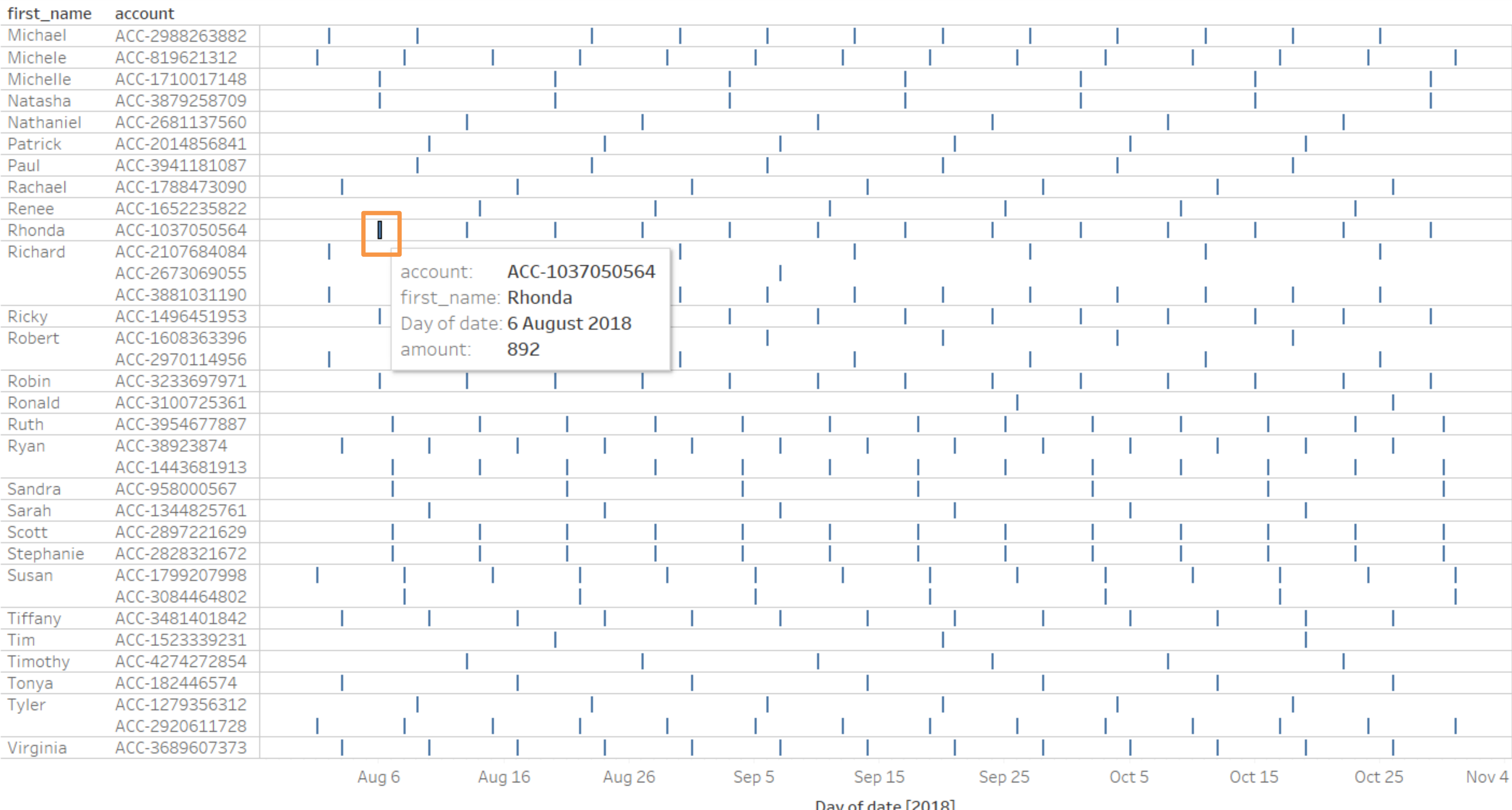
ANZ VIRTUAL INTERNSHIP
TASK2

BY: TANISHA JAIN
GITHUB: https://github.com/tanjaingit

ANZ DATASET ANALYSIS

- *The dataset contains information of 100 hypothetical customers for 3 months.*
- *Below distribution shows pay and paycounts of customers (in detail) from 29July to 30 Oct.*
- *I observed that pay for each customer is constant. However, paycount varies from 1 – 14. Hence, I decided to calculate annual salaries by grouping customers acc. to their paycounts.*

PAY DISTRIBUTION FOR EACH CUSTOMER



DATA PREPARATION

- *I calculated annual salaries of customers based on 3 categories:*
  a) Weekly b) Fortnightly c) Monthly

- *Attributes used for prediction are gender, age, balance, amount and spendings*

- *Here, spendings is a feature engineered attribute calculated by adding all debit transactions.*

CALCULATION OF ANNUAL SALARY FOR EACH CUSTOMER

```
df_acc["annual_salary"] = 0
for i in range(0,len(df_acc.pay_count)):
    #weekly pay
    if df_acc["pay_count"][i] >=12:
        df_acc["annual_salary"][i] = df_acc["pay"][i] / 7 *365.25
    #monthly pay
    elif df_acc["pay_count"][i] <=5:
        df_acc["annual_salary"][i] = df_acc["pay"][i] * 12
    #fortnightly pay
    else:
        df_acc["annual_salary"][i] = df_acc["pay"][i] / 14 *365.25

df_acc.head()
```

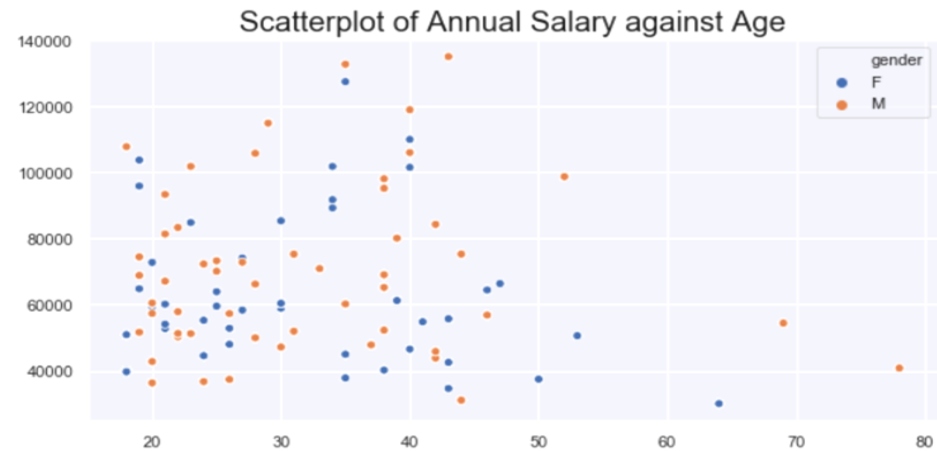
SELECTED ATTRIBUTES TO FIND CORRELATION

	gender	balance	age	amount	annual_salary	spendings
0	F	1735.120675	26	45.348772	52856	12020.21
1	F	1735.120675	26	45.348772	52856	12020.21
2	M	1191.291419	38	78.206106	52282	10668.76
3	F	3331.424479	40	74.465019	46543	7689.27
4	F	1735.120675	26	45.348772	52856	12020.21

FEATURE ENGINEERING OF spendings ATTRIBUTE

```
# feature engineering of debit attribute
df_acc['spendings'] = df_acc['POS']+ df_acc['SALES-POS']+df_acc['PAYMENT']+
                     df_acc['INTER BANK']+df_acc['PHONE BANK']
```

CORRELATION

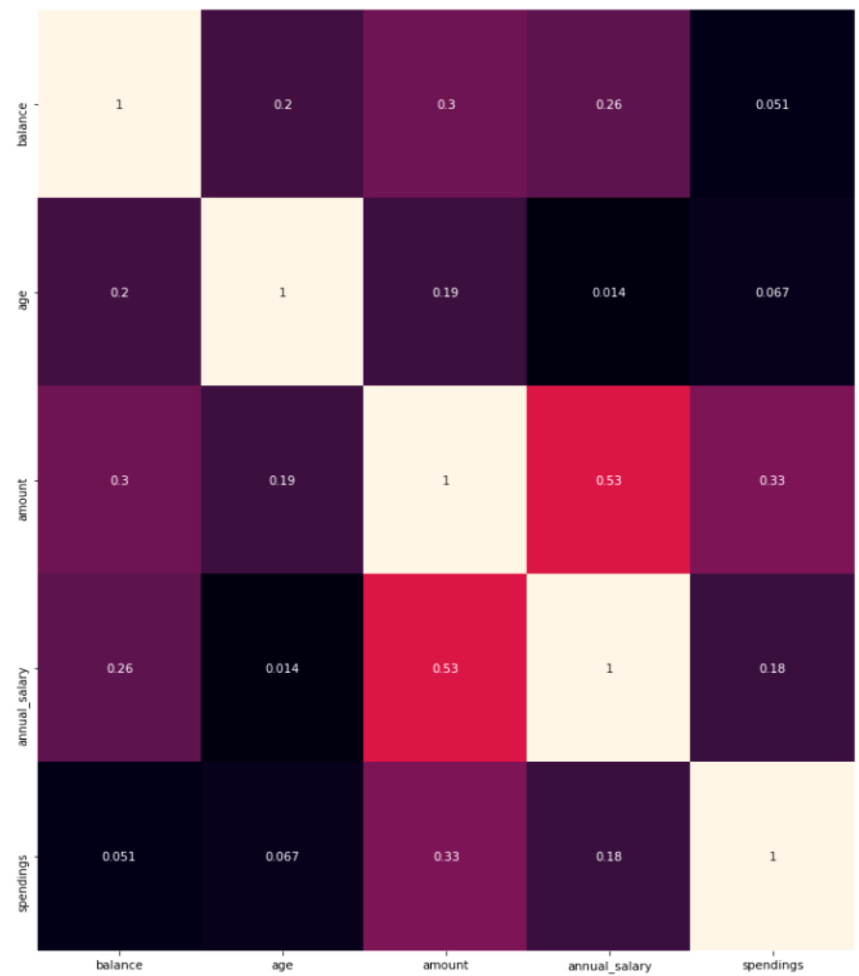


- *annual_salary feature has significant correlations spendings.*

- *balance and spendings(feature engineered) also correlate to annual_salary, but lesser than amount.*

- *age cannot be used for prediction as its correlation is not significant.*

- *Data points on scatter plot do not show any pattern. F-gender cannot be well distinguished with M-gender. Hence, gender cannot be used for prediction.*



LINEAR REGRESSION

- *Model used: LinerRegression from sklearn package.*
- *The accuracy score of this model is 0.67 which is not good.*
- *The scatter plot shows how well the model is doing. The variance between actual values and predicted values is very high.*
- *R2 score = 0.67*

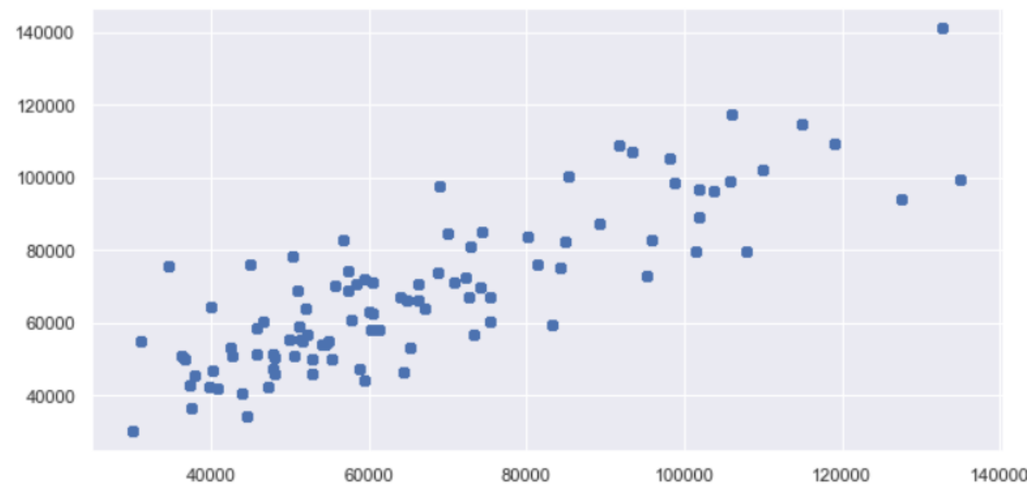
LINEAR REGRESSION MODEL FIT – PREDICT ANNUAL SALARY

```python
#LINEAR REGRESSION
from sklearn.linear_model import LinearRegression
# Setting up random seed
np.random.seed(42)
# Instantiate the model
model_lin = LinearRegression()
# Fit the model
model_lin.fit(X_train , y_train)
# Making predictions
y_lin_preds = model_lin.predict(X_test)
# Model Score
model_lin.score(X_test , y_test)
```

ACTUAL VALUES (y_test) Vs PREDICTED VALUES (y_lin_preds)

```python
#Visualizing with scatter plot how well our model is doing
plt.scatter(y_test , y_lin_preds)
```

```
<matplotlib.collections.PathCollection at 0x235d91dada0>
```



DECISION TREE

- *Model used: DecisionTreeRegressor from sklearn package.*
- *I used percentile scores for decision tree regression*
- *The accuracy score of this model is 0.75 which is not good.*
- *The scatter plot shows how well the model is doing. The variance between actual values and predicted values is very high.*
- *R2 score = 0.75*

DECISION TREE BASED MODEL FIT–PREDICT ANNUAL SALARY

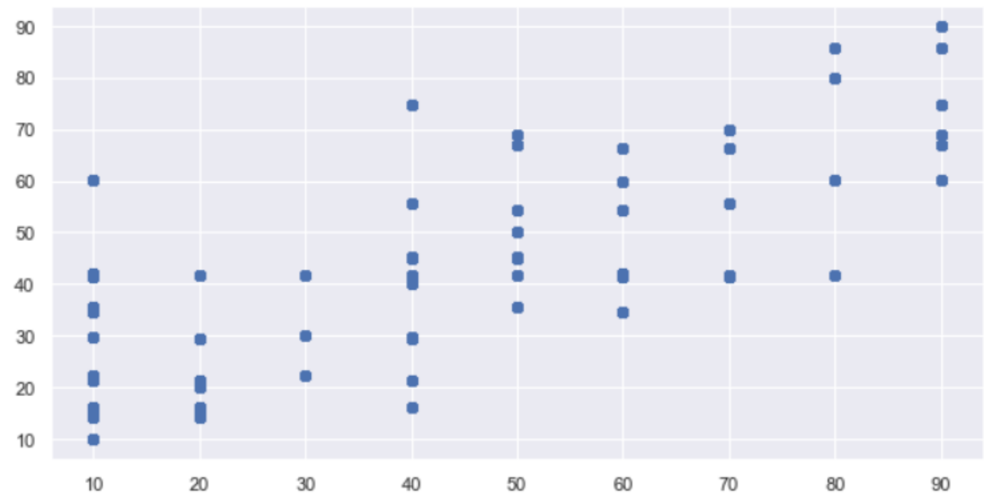
```
# Create Decision Tree classifer object
from sklearn.tree import DecisionTreeRegressor
np.random.seed(42)
# Instantiate the model
model_reg = DecisionTreeRegressor(max_depth = 9)
model_reg.fit(X_train , y_train)
# Score of the model
model_reg.score(X_test , y_test)
# Make predictions
y_preds = model_reg.predict(X_test)
# Checking the score
model_reg.score(X_test , y_test)
```

```
0.7588109509796269
```

ACTUAL VALUES (y_test) Vs PREDICTED VALUES (y_preds)

```
#Visualizing with scatter plot how well our model is doing
plt.scatter(y_test , y_preds)
```

```
<matplotlib.collections.PathCollection at 0x1e307538940>
```



SUMMARY

- *The features selected did not produce a good model to predict annual salary of the customers.*
- *As we can see that accuracy score of linear regression is 0.67 and decision tree regression is 0.75 which are very low. Both the models are inaccurate. (However accuracy score should never be used as a standard test for models)*
- *We can see that models shoe high variance as well as high bias.*
- *Generally , the companies that customers work for ; years of experience + initial salary ; type of job (technical/ business/ management) are better features to use as salary predictors.*

LINEAR REGRESSION

	Actual values	Predicted values	Differences
8457	91756	107472.870769	15716.870769
3414	51656	59362.154032	7706.154032
7606	59379	50759.860068	-8619.139932
2686	80120	74094.625534	-6025.374466
4317	37385	41412.093814	4027.093814

DECISION TREE

	Actual values	Predicted values	Differences
4199	30	30.000000	0.000000
8533	80	80.000000	0.000000
10372	50	35.660377	-14.339623
11185	10	41.854839	31.854839
8146	20	20.000000	0.000000

COMPARING RESULTS FROM BOTH MODELS

```
Decision Tree Model:
R2 Score:75.881095%
Mean Squared Error:179.284066
Mean Absolute Error:8.154801
.........................................
Linear Regression Model:
R2 score:  67.49799511907143 %
Mean Squared error:  176699172.45333818
Mean absoulte error:  10137.38751106627
```